Neural Local Wasserstein Regression

Inga Girshfeld
Department of Mathematics
University of Southern California
Los Angeles, CA 90007
girshfel@usc.edu

Xiaohui Chen
Department of Mathematics
University of Southern California
Los Angeles, CA 90007
xiaohuic@usc.edu

Abstract

We study the estimation problem of distribution-on-distribution regression, where both predictors and responses are probability measures. Existing approaches typically rely on a global optimal transport map or tangent-space linearization, which can be restrictive in approximation capacity and distort geometry in multivariate underlying domains. In this paper, we propose the *Neural Local Wasserstein Regression*, a flexible nonparametric framework that models regression through locally defined transport maps in Wasserstein space. Our method builds on the analogy with classical kernel regression: kernel weights based on the 2-Wasserstein distance localize estimators around reference measures, while neural networks parameterize transport operators that adapt flexibly to complex data geometries. This localized perspective broadens the class of admissible transformations and avoids the limitations of global map assumptions and linearization structures. We develop a practical training procedure using DeepSets-style architectures and Sinkhorn-approximated losses, combined with a greedy reference selection strategy for scalability. Through synthetic experiments on Gaussian and mixture models, as well as distributional prediction tasks on MNIST, we demonstrate that our approach effectively captures nonlinear and high-dimensional distributional relationships that elude existing methods.

1 Introduction

Modeling relationships where both predictors and responses are probability measures is an emerging challenge in statistics and machine learning. Such *distribution-on-distribution* (DoD) regression arises naturally in diverse applied domains where data are inherently measure-valued. In computer vision, images can be processed as two-dimensional histograms of grayscale pixel levels (or in the RGB space for colored images), and the task of super-resolution is to constructing a high-resolution image as the response distribution from a given low-resolution image as the predictor [Kim and Kwon, 2010, Lai et al., 2017]. In biomedical sciences, population heterogeneity such as mortality rate is often captured through empirical age-at-death distributions [Chiou and Müller, 2009, Shang and Hyndman, 2017]. In fluid dynamics, researchers routinely compare and predict distributions of spatiotemporal fields such as temperature and precipitation in climate sciences [Jiang et al., 2020, Li et al., 2021]. Across these settings, regression directly in the space of measures provides a principled framework for capturing complex distributional relationships that methods designed for Euclidean data cannot accommodate.

A central difficulty is that probability measures reside in a non-Euclidean space. A large body of recent work has sought to endow such spaces with meaningful geometry through the theory of optimal transport (OT) [Villani, 2009, Santambrogio, 2015]. One line of work defines regression through *global* OT maps. [Ghodrati and Panaretos, 2022a,b, 2023] formalizes DoD regression by transporting the predictor distribution via a Monge map, with minimax analysis and Gaussian

1st Conference on Topology, Algebra, and Geometry in Data Science (TAG-DS 2025).

extensions. [Okano and Imaizumi, 2024] further specializes this framework to Gaussian families via the Bures–Wasserstein geometry. An alternative direction leverages *tangent-space linearization*, notably Wasserstein regression [Chen et al., 2023], which lifts measures to tangent bundles, performs regression in linearized coordinates, and maps predictions back. More broadly, the Fréchet regression framework [Petersen and Müller, 2019] provides kernel-based estimators for random objects, with inference tools such as Wasserstein F-tests on Bures–Wasserstein manifolds [Xu and Li, 2025]. A third strand concerns *conditional optimal transport*, where maps depend on covariates: data-driven conditional OT [Tabak et al., 2021], conditional Brenier's map via entropic regularization [Baptista et al., 2024], and neural conditional maps [Wang et al., 2023].

While these approaches provide important foundations, they share key limitations. Global-map methods [Oliva et al., 2013, Ghodrati and Panaretos, 2022a,b, 2023] impose strong structural assumptions and are difficult to scale beyond one-dimensional or Gaussian data. Tangent-space methods [Chen et al., 2023, Petersen and Müller, 2019] depend on linearization at the Fréchet means of predictor and response measures, which can distort geometry in higher dimensions than the univariate case. Conditional OT [Tabak et al., 2021, Baptista et al., 2024, Wang et al., 2023] defines maps indexed by covariates but still assumes each map is globally defined. To the best of our knowledge, no prior DoD regression framework explicitly models the transport as *locally defined* over subsets of the source measure's support.

In this paper, we introduce a new framework for nonparametric Wasserstein regression where both source and target are probability measures. Our model departs from the prevailing paradigms by allowing the transport between measures to be only *locally* defined, rather than enforcing a single global map or a linearized approximation. This local perspective not only broadens the class of admissible transformations (cf. Section 3.1 for the counterexample of the global map) but also offers robustness in higher-dimensional settings where global maps may fail to exist or be statistically unstable.

1.1 Our Contribution

We introduce a new framework for modeling nonparametric Wasserstein regression where both source and target are probability measures. We propose the *neural local Wasserstein regression*, a statistical framework for DoD regression based on locally defined transport maps that generalizes beyond one-dimensional or Gaussian data and avoids restrictive global-map assumptions and linearization structures. We demonstrate the flexibility of our method on synthetic and real examples and illustrate how local-map regression can capture complex higher-dimensional distributional relationships.

2 Background

2.1 Wasserstein Distance and Optimal Transport

Let $\mathcal{P}_2(\mathbb{R}^d)$ denote the space of Borel probability measures on $\mathbb{R}^d$ with finite second moment. The squared *2-Wasserstein distance* between $\mu, \nu \in \mathcal{P}_2(\mathbb{R}^d)$ is defined as

$$W_2^2(\mu,\nu) = \inf_{\pi \in \Pi(\mu,\nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x-y\|^2 \, d\pi(x,y),$$

where $\Pi(\mu,\nu)$ is the set of couplings with marginals μ and ν. This defines a metric on $\mathcal{P}_2(\mathbb{R}^d)$ that encodes both distributional and geometric information [Villani, 2009, Santambrogio, 2015]. When μ is absolutely continuous with respect to Lebesgue measure, the *Brenier theorem* guarantees that there exists a unique optimal transport map $T : \mathbb{R}^d \to \mathbb{R}^d$ such that $T_\sharp \mu = \nu$ and

$$W_2^2(\mu,\nu) = \int_{\mathbb{R}^d} \|x - T(x)\|^2 \, d\mu(x),$$

and this map is characterized as the gradient of a convex potential [Brenier, 1991]. These properties make the Wasserstein space a Riemannian-like metric space, with geodesics given by displacement interpolation [Ambrosio et al., 2008, Panaretos and Zemel, 2020]. The Wasserstein distance has become a fundamental tool in statistics and machine learning, providing a principled way to compare and model distributions in fields ranging from generative modeling to functional data analysis [Peyré and Cuturi, 2019, Panaretos and Zemel, 2020]. In the context of regression, it allows us to directly model relationships between probability measures as geometric objects.

2.2 Classical Nonparametric Regression

In Euclidean settings, regression between random variables $Y \in \mathbb{R}$ and predictors $X \in \mathbb{R}^d$ has been extensively studied. A cornerstone is *nonparametric regression*, where the regression function $m(x) = \mathbb{E}[Y \mid X = x]$ is estimated without parametric assumptions. One widely used class of estimators is *kernel smoothing*. The Nadaraya–Watson estimator is defined as

$$\hat{m}(x) = \frac{\sum_{i=1}^{n} K_h(x - X_i) Y_i}{\sum_{i=1}^{n} K_h(x - X_i)}, \tag{1}$$

where $K_h(u) = h^{-d} K(u/h)$ is a kernel function with bandwidth h. Under mild smoothness assumptions, kernel estimators achieve the minimax-optimal convergence rates for nonparametric regression [Tsybakov, 2009]. Beyond kernel methods, local polynomial estimators provide bias reduction while retaining similar variance properties [Fan and Gijbels, 1996]. These methods form the classical toolkit for nonparametric regression, and they inspire our extension to regression in the space of probability measures (cf. Section 3.2 below). The analogy is that, just as kernel smoothing locally averages Euclidean responses, one may seek analogous "local averaging" or "local transport" operations in Wasserstein space to estimate regression functions between distributions.

3 Methodology

In this section, we first introduce a nonparametric analog of the standard regression model $Y = f(X) + \varepsilon$ in the Euclidean space where $f : \mathbb{R}^d \to \mathbb{R}$ is the regression function of interest and ε is a mean-zero noise term. Then, we present our local kernel smoothing estimator based on the Wasserstein geometry. After that, we will describe the neural network architecture that will be used to solve the localized transportation maps for the nonparametric Wasserstein regression.

3.1 Wasserstein Nonparametric Regression Model

Let $\mu \in \mathcal{P}_2(\mathbb{R}^d)$ be a source probability measure (i.e., the reference measure). We define the nonparametric regression model operating between the source and target Wasserstein spaces as

$$\nu = T_\varepsilon \# (T_\mu \# \mu), \tag{2}$$

where $T_\mu(x) \coloneqq T(\mu, x) : \mathcal{P}_2\left(\mathbb{R}^d\right) \times \mathbb{R}^d \to \mathbb{R}^d$ is the signal map at μ and $T_\varepsilon : \mathbb{R}^d \to \mathbb{R}^d$ represents the random noise perturbation satisfying the *unbiasedness* criterion, i.e., the mean-preserving condition

$$\mathbb{E}\left[T_\varepsilon(x)\right] = x, \quad \forall x \in \mathbb{R}^d.$$

Here, $\#$ is the pushforward operation. In what follows, we focus on estimating the signal map T_μ at any given source measure μ from observed data where $\{(\mu_i, \nu_i)\}_{i=1}^n$ are independently sampled from model 2 via $\nu_i = T_{\varepsilon_i} \# (T_{\mu_i} \# \mu_i)$ with i.i.d. sampled noise maps T_{ε_i} for $i = 1, \ldots, n$.

Our proposed model 2 is different from the regression model considered in the previous works [Oliva et al., 2013, Ghodrati and Panaretos, 2022a, 2023], where the regression operator is defined as

$$\nu = T_\varepsilon \# (T_0 \# \mu), \tag{3}$$

for some unknown transport map $T_0 : \mathbb{R}^d \to \mathbb{R}^d$. Note that the signal map T_0 in 3 does not depend on the input source measure μ. Thus, it is a direct analogue of the Euclidean scalar-on-vector regression by viewing the transport map T_0 as the "regression function" and is only capable of capturing the global relationship between μ and ν.

Clearly, model 3 is a special case of (and thus less flexible than) our proposed model 2 since the latter allows the transport map to depend and reflect some features from the source domain. A simple counterexample that model 3 cannot cover is when both source and target distributions μ and ν are discrete over $\mathbb{R}$ supported at two points 0 and 1. Then model 3 implies that either $\nu(1) = \mu(1)$ or $\nu(1) = \mu(0)$, which does not allow $\nu(1)$ to be as a general function of $\mu(1)$ or $\mu(0)$. In contrast, our model 2 can be viewed as regressing the optimal transport map T_μ from μ to ν (as the response) on μ as the covariate.

We emphasize that the proposed model 2 is more flexible than operator learning approaches [Gracyk and Chen, 2024, Amos et al., 2023], which aim to learn the mapping from source to target measures.

In our setting, although the noise maps T_{ε_i} are i.i.d., the sampling model 2 produces data pairs $\{(\mu_i, \nu_i)\}_{i=1}^n$ that are independent but *not identically distributed*. Moreover, the covariate-adjusted interpretation of the local map T_μ naturally entails distribution shift from training data when making predictions on unseen μ, a phenomenon that neural operator models are not equipped to handle.

3.2 Local Smoothing Estimator

Now, we consider a supervised learning procedure to estimate the transport maps $\{T_\mu\}_{\mu \in \mathcal{P}_2(\mathbb{R}^d)}$ from pairs of the observed data $\{(\mu_i, \nu_i)\}_{i=1}^n$. Let us fix a reference measure μ, and our goal is to learn the map T_μ. Our starting point is the well-known variational formulation of the classical kernel smoothing estimator 1 as

$$\hat{m}(x) := \arg\min_{c \in \mathbb{R}} \sum_{i=1}^n (c - Y_i)^2 K\Big(\frac{\|x - X_i\|}{h}\Big), \tag{4}$$

where $\|\cdot\|$ is the Euclidean norm on $\mathbb{R}^d$. Note that the variational form only requires the input features/predictors to stay in a metric space (i.e., a vector space structure is not needed). Thus, 4 can be naturally extended to a local smoothing estimator under the Wasserstein geometry $(W_2, \mathcal{P}_2(\mathbb{R}^d))$. Specifically, we propose to minimize the following empirical loss function as

$$\hat{T}_\mu := \arg\min_{T:\mathbb{R}^d \to \mathbb{R}^d} \left\{ \mathcal{L}_n(T) := \frac{1}{n} \sum_{i=1}^n W_2^2 \left(T \# \mu_i, \nu_i\right) K_h(\mu, \mu_i) \right\}, \tag{5}$$

where $K_h(\mu_1, \mu_2) = h^{-d} K(W_2(\mu_1, \mu_2)/h)$ and $K(u)$ is a kernel function satisfying $K(u) \geq 0$ and $\int_{\mathbb{R}} K(u) du = 1$. The kernel function $K_h(\mu, \mu_i)$ plays a central role in localizing the estimator around a reference measure μ. It down-weights the influence of training pairs (μ_i, ν_i) that are far from μ under the 2-Wasserstein distance, ensuring that the estimated map $\hat{T}_\mu$ reflects the local geometry of the regression problem. This design promotes flexibility in learning distribution-to-distribution relationships that may vary across the space of input measures.

Making analogy to the local linear estimators for classical nonparametric regression in the Euclidean space [Fan and Gijbels, 1996], if we parameterize $T_\mu(x) = \alpha(\mu) + B(\mu)x$, then we obtain a Wasserstein local linear regression estimator via 5. On the other hand, since the local linear regression has the curse-of-dimensionality [Tsybakov, 2009] and our primary goal is to model the transport maps between two infinite-dimensional source and target Wasserstein spaces, we adopt a neural network approach as our functional approximator for the local regression operators (cf. Section 3.4 for more details).

Once trained, each local estimator $\hat{T}_\mu$ can be reused at test time to make predictions for new source distributions μ' that are close to μ, by applying the learnt map $\hat{T}_\mu$ to samples from μ'. In practice, we train multiple local models centered at a collection of reference measures $\{\mu_0^{(l)}\}_{l=1}^M$, and for a given test distribution, predictions can be made using the nearest $\mu_0^{(l)}$ or a weighted combination of nearby local estimators. This localized approach allows us to approximate complex global regression structures using a set of simpler, interpretable affine transformations.

3.3 Kernel Bandwidth Tuning

The hyperparameter, specific to our framework, that required tuning, aside from standard choices such as learning rate or batch size, was the kernel bandwidth h. Across all experiments (Gaussian, Gaussian mixture, and MNIST in Section 4), we employed a nearest-neighbor heuristic to select h in a data-adaptive manner. Specifically, let $\{\mu_i\}_{i=1}^n$ denote the training source distributions, and fix a test distribution μ_0, and compute pairwise distances. For a set number of neighbors $k \ll n$, let

$$r_k(\mu_0) \;=\; \min \left\{ r : \sum_{i=1}^n \mathbf{1}\left\{i : d(\mu_0, \mu_i) \leq r\right\} \geq k \right\},$$

the distance to the k-th nearest neighbor of μ_0. We then set the kernel bandwidth

$$h(\mu_0) \;=\; \rho\, r_k(\mu_0),$$

with scaling constant $\rho > 0$. Typically $\rho = 1$, but is adapted as the dimension increases to account for the curse of dimensionality.

In words, the bandwidth is chosen so that the effective kernel support includes approximately k training distributions around μ_0. This ensures that the regression estimator is smoothed appropriately while adapting to the local density of training distributions in Wasserstein space, even in settings when the data is highly irregular. The same principle was used across all simulation settings with slight modifications to adjust to differences in Gaussian, Gaussian Mixture, and MNIST data.

3.4 Neural Network Architectures

In this section, we describe how these estimators are learned in practice using a neural networks tailored to each setting. We utilize two different neural net architectures, depending on the task. For source and target pairs of empirical distributions, we employ a DeepSets neural network architecture Zaheer et al. [2018] to handle the permutation invariant nature of the given data. Moreover, we employ U-Nets Ronneberger et al. [2015] as the task becomes an image-to-image regression learning task.

3.4.1 DeepSets

To estimate local transport maps, we adopt a nonparametric, local regression framework in which a separate neural network is trained for each chosen reference distribution $\mu_0^{(l)}$, using only training pairs that are close to $\mu_0^{(l)}$ in the Wasserstein sense. Each network learns to output a transport map $T : \mathbb{R}^d \to \mathbb{R}^d$ in one of two variants:

$$T(x) = \alpha + Bx, \qquad T(x) = x + \Delta(x, z),$$

where (α, B) parameterize an affine transformation and $\Delta(x, z)$ denotes a more general local map, where the network conditions on the distribution via a global context vector z, and predicts per-point displacements. The learning objective is a Sinkhorn-approximated Wasserstein-2 loss, weighted by the kernel $K_h(\mu_0^{(l)}, \mu_i)$.

Let μ_i and ν_i denote source and target distributions, respectively, and suppose we observe k i.i.d. samples from each,

$$x_j^{(i)} \overset{\text{iid}}{\sim} \mu_i, \qquad y_j^{(i)} \overset{\text{iid}}{\sim} \nu_i, \qquad j = 1, \ldots, k.$$

We define empirical measures

$$\hat{\mu}_i = \frac{1}{k} \sum_{j=1}^{k} \delta_{x_j^{(i)}}, \qquad \hat{\nu}_i = \frac{1}{k} \sum_{j=1}^{k} \delta_{y_j^{(i)}}.$$

Each network uses a DeepSets-style architecture [Zaheer et al., 2018] to process the input distribution $\hat{\mu}_i$ by its point cloud representation $X_i = \{x_1^{(i)}, \ldots, x_k^{(i)}\} \subset \mathbb{R}^d$. The encoder–decoder structure is

$$f_l(X_i) = \rho_l \left(\frac{1}{k} \sum_{j=1}^{k} \psi_l(x_j^{(i)}) \right),$$

where $\psi_l : \mathbb{R}^d \to \mathbb{R}^h$ is a pointwise encoder (two-layer MLP with ReLU), the sum produces a permutation-invariant aggregation, and ρ_l is a decoder whose output depends on the chosen map variant.

In the affine case, the decoder outputs

$$(\alpha_i^{(l)}, B_i^{(l)}) = \rho_l(z_i), \quad \alpha_i^{(l)} \in \mathbb{R}^d, \quad B_i^{(l)} \in \mathbb{R}^{d \times d}.$$

In the general local map case, the decoder produces a global context vector z_i, which is concatenated with each point x and passed through an additional *displacement head* to compute per-point corrections $\Delta(x, z_i)$.

Thus, for each reference distribution $\mu_0^{(l)}$, the network learns a local model $\hat{T}_{\mu_0^{(l)}}$ such that $\hat{T}_{\mu_0^{(l)}} \# \hat{\mu}_i \approx \hat{\nu}_i$. This procedure is identical whether the distributions are Gaussians, Gaussian mixtures, or empirical distributions from real data, since in all cases the model only sees their sampled point clouds and the architecture is inherently permutation-invariant.

3.4.2 U-Net

For distributions supported on regular image grids (e.g. MNIST digits), we employ a compact U-Net architecture [Ronneberger et al., 2015], which is natural for image-to-image regression tasks. Unlike the permutation-invariant DeepSets model, the U-Net exploits the underlying spatial structure of the data through convolutions and skip connections. Let $x \in \mathbb{R}^{1\times H\times W}$ denote a grayscale input image. The network $f_\theta : \mathbb{R}^{1\times H\times W} \to \mathbb{R}^{1\times H\times W}$ follows the schematic form

$$x \xrightarrow{E_1} s_1 \xrightarrow{E_2} s_2 \xrightarrow{M} z \xrightarrow{\text{upsample}\times 2} \tilde{z} \xrightarrow{[\tilde{z};\, s_1]} D_1 \xrightarrow{\text{output}} f_\theta(x),$$

where $[\cdot;\cdot]$ denotes channel-wise concatenation.

E_1 and E_2 are the encoder blocks, progressively reducing spatial resolution while increasing the number of feature channels, M is the bottleneck at the coarsest resolution, the upsampling stage restores spatial resolution, and D_1 is the decoder block that fuses the upsampled features $\tilde{z}$ with the high-resolution skip connection s_1. The output head then maps the decoder features to a prediction $f_\theta(x)$.

This U-Net is lightweight, with a single downsampling stage, one bottleneck, and a symmetric upsampling path. Skip connections link encoder and decoder features at matching resolutions, enabling the model to combine local detail (s_1) with global context (z). Convolutions are followed by normalization and nonlinear activations, and residual connections are used within blocks to stabilize training. The final layer outputs an image of the same resolution as the input, which we interpret as a probability distribution after applying a softplus transform and per-image normalization.

4 Simulation Studies

We conducted experiments using three types of datasets, Gaussian, Gaussian mixture, and MNIST image pairs. Model hyperparameters, include learning rate, batch size, number of epochs, kernel bandwidth h (discussed in Section 3.3), and the blur parameter ε associated with the Sinkhorn approximation used to estimate the Wasserstein distance. All models were trained using the Adam optimizer with default momentum parameters, a learning rate of 10^{-3}, and the blur parameter was fixed $\varepsilon = 0.15$, unless otherwise noted. Moreover, evaluation was performed on unseen test pairs.

4.1 Gaussian Experiments

We trained a family of local models to pushforward source distributions μ_i toward their targets ν_i, using a kernel-weighted Sinkhorn loss centered at several reference distributions $\mu_0^{(l)}$. Details for the data generation are available in Sec A.1.1. Each local model was trained independently using only those training pairs (μ_i, ν_i) that were close, in the 2-Wasserstein sense, to a given reference $\mu_0^{(l)}$. Note, in later experiments, we change the strategy slightly.

In these early experiments, when evaluating the performance of our methods on simple Gaussian distributions, we found that even simple affine maps $T(x) = Bx + \alpha$ were capable of producing good visual alignment between the pushed-forward source distribution and the target as seen in Figure [1]. We trained the neural network introduced in Section 3.4.1 to learn the parameters B and α.

4.2 Gaussian Mixtures Experiments

In the Gaussian mixture experiments, the estimator is constructed in a local fashion. Given a fixed test source distribution μ_0 with corresponding target ν_0, and a collection of training pairs $\{(\mu_i, \nu_i)\}_{i=1}^n$, the goal is to learn a local transport map $\hat{T}_{\mu_0}$ such that

$$\hat{T}_{\mu_0} \# \mu_0 \;\approx\; \nu_0.$$

The estimator is trained using only those training pairs (μ_i, ν_i) that lie within the effective support of the kernel centered at μ_0, with weights $K_h(\mu_0, \mu_i)$. For this setting, we use the generalized DeepSets architecture from Section 3.4.1 that outputs an empirical prediction $\hat{\nu}_0$ directly from the point cloud representation of μ_0 and its kernel-weighted neighbors.

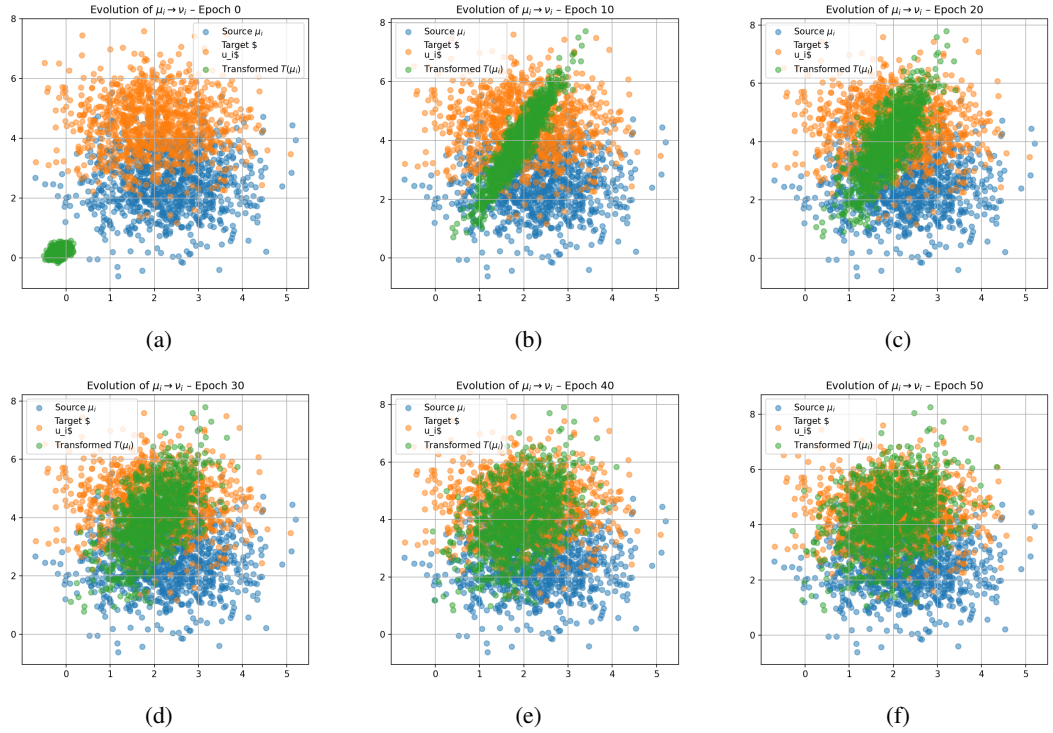

Figure 1: Progression of pushforward distributions for a fixed reference measure $\mu_0^{(0)}$ over training epochs.

The description for the generated data used in this setting is available in Section A.1.2. We store the training data in tensors

$$\texttt{mu} \in \mathbb{R}^{N\times K\times 2}, \qquad \texttt{nu} \in \mathbb{R}^{N\times K\times 2},$$

as a *master dataset* (e.g., $N = 1000$, $K = 10{,}000$). Training regimes (n, k) are created by deterministic subsampling using a subset seed. Each Regime is run 10 times by subsampling the master dataset with a different seed.

Let $\{(\mu_i, \nu_i)\}_{i=1}^n$ be training pairs, and let $\hat{T}_{\mu_0}$ denote the estimated local transport map used to predict ν_0 from μ_0. Define the empirical Wasserstein barycenter of the target measures as

$$\bar{\nu}_W = \arg\min_{\nu\in\mathcal{P}_2(\mathbb{R}^d)} \sum_{i=1}^{n} W_2^2(\nu, \nu_i).$$

Then, we define $SS_{\text{res}} = W_2^2\big(\nu_0, \hat{T}_{\mu_0}\#\mu_0\big)$, $SS_{\text{tot}} = W_2^2\big(\nu_0, \bar{\nu}_W\big)$, and the *Wasserstein coefficient of determination* as

Table 1: Average test error and Wasserstein coefficient of determination R_W^2 (Eq. 6) across simulation regimes.

d	n	k	Absolute Test Error	R_W^2
2	100	1000	0.01922 ± 0.00694	0.99977
2	100	100	0.02438 ± 0.00995	0.99971
2	10	1000	0.68470 ± 0.34196	0.99190
2	1000	1000	0.00124 ± 0.00011	0.99983
5	100	1000	2.34761 ± 0.03216	0.94945
5	100	100	2.36256 ± 0.03350	0.94913
5	100	10	23.07784 ± 1.30152	0.94064
5	10	100	19.52894 ± 3.72827	0.95291

$$R_W^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}} = 1 - \frac{W_2^2(\nu_0, \hat{T}_{\mu_0}\#\mu_0)}{W_2^2(\nu_0, \bar{\nu}_W)}. \tag{6}$$

In the two-dimensional setting, our estimator achieves near-exact recovery provided sufficient sample size ($n \geq 100$), with relative errors on the order of 10^{-4}. Performance deteriorates rapidly with small sample size ($n = 10$), exacerbated by insufficient kernel support. In higher dimension ($d = 5$), both absolute and relative errors increase by one to two orders of magnitude, with relative errors

saturating around 0.65–0.70. To keep results consistent, the bandwidth used in the two-dimensional cases was recycled in the five-dimensional cases, only scaled by a factor to adjust for the increase in dimension. Thus, it is important to note that the training could have potentially been optimized for the five-dimensional case. The results indicate possible sensitivity to the curse of dimensionality and highlight the importance of appropriate bandwidth scaling. Overall, these results demonstrate that the method is highly effective in low-dimensional regimes with adequate data, but requires careful tuning and larger sample sizes to remain stable as dimension increases.

4.3 MNIST Experiment

In this experiment, we evaluate our method on the benchmark MNIST dataset of handwritten digits, constructing a subset designed to test distributional regression in a heterogeneous setting. Specifically, we select five digit pairs: $(0 \to 5), (1 \to 7), (2 \to 8), (3 \to 6), (4 \to 9)$. For each pair, all images of the source digit serve as source distributions $\{\mu_i\}_{i=1}^n$, while the corresponding images of the target digit serve as target distributions $\{\nu_i\}_{i=1}^n$. From each pair we reserve one source–target example (μ_0^a, ν_0^b) for testing, and use the remainder for training. The U-Net described in Section 3.4.2 is trained to learn local maps $T_{\mu_0^a}$, which are then evaluated by predicting

$$\nu_0^b \;\approx\; T_{\mu_0^a} \# \mu_0^a,$$

for each reserved test pair.

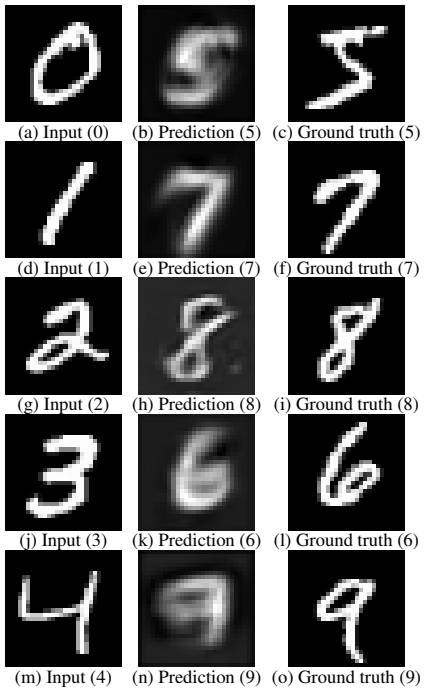

Figure 2: MNIST transformation results across multiple digit pairs. Each row shows a source digit (left), predicted output (middle), and ground-truth target (right). Images are treated as normalized discrete probability distributions while evaluating loss.

In all experiments we used $N_{\text{per class}} = 1000$ samples. Each 28×28 grayscale image was mapped to the unit square $[0,1]^2$ by assigning normalized pixel intensities to the centers of the pixels, thereby representing each image as a discrete probability distribution. The Wasserstein kernel loss was then computed directly on these empirical measures. We made a strong assumption that all images of one type would be closer to each other in the Wasserstein sense despite a large variation in how images are handwritten. This proved to hold in all but one case, and, in that case, we trimmed some percentages off entirely to concentrate the signal.

5 Discussion and Limitations

This work introduces *Neural Local Wasserstein Regression*, a nonparametric DoD regression framework that models transport via locally defined maps, trained with kernel weights in W_2 space and instantiated with permutation-invariant (DeepSets) and convolutional (U-Net) architectures. The results suggest that local transport improves approximation capacity relative to global map or linearization-based approaches when the predictor space exhibits heterogeneous geometry. Practically, local training reduces the reliance on a single global map that may not exist or be stable, enables reuse of fitted maps in the vicinity of a reference measure, and supports modular deployment (multiple local models can be trained and composed or selected at test time).

Local estimators rely on the kernel bandwidth h and the effective number of neighbors. While our kNN heuristic adapts to local density, it remains a tuning knob. Key questions about identifiability of T_μ, consistency and rates under local smoothness of the transport field, and the effect of h on statistical and computational error will be pursued in future work.

References

Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient Flows in Metric Spaces and in the Space of Probability Measures*. Birkhäuser, 2008.

Brandon Amos, Giulia Luise, Samuel Cohen, and Ievgen Redko. Meta optimal transport. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Ricardo Baptista, Aram-Alexandre Pooladian, Michael Brennan, Youssef Marzouk, and Jonathan Niles-Weed. Conditional simulation via entropic optimal transport: Toward nonparametric estimation of conditional Brenier maps. *arXiv preprint arXiv:2411.07154*, 2024.

Yann Brenier. Polar factorization and monotone rearrangement of vector-valued functions. *Communications on Pure and Applied Mathematics*, 44(4):375–417, 1991.

Yaqing Chen, Zirong Lin, and Hans-Georg Müller. Wasserstein regression. *Journal of the American Statistical Association*, 2023.

Jeng-Min Chiou and Hans-Georg Müller. Modeling hazard rates as functional data for the analysis of cohort lifetables and mortality forecasting. *Journal of the American Statistical Association*, 104 (486):572–585, 2009. doi: 10.1198/jasa.2009.0023. URL `https://doi.org/10.1198/jasa.2009.0023`.

Jianqing Fan and Irene Gijbels. *Local Polynomial Modelling and Its Applications*. Chapman and Hall/CRC, 1996.

Laya Ghodrati and Victor M. Panaretos. Distribution-on-distribution regression via optimal transport maps. *Biometrika*, 109(4):957–974, 2022a.

Laya Ghodrati and Victor M. Panaretos. Minimax rate for optimal transport regression between distributions. *arXiv preprint arXiv:2206.01447*, 2022b.

Laya Ghodrati and Victor M. Panaretos. Transportation of measure regression in higher dimensions. *arXiv preprint arXiv:2305.17503*, 2023.

Andrew Gracyk and Xiaohui Chen. GeONet: a neural operator for learning the Wasserstein geodesic. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, July 2024. doi: 10.48550/ARXIV.2209.14440.

Chiyu "Max" Jiang, Soheil Esmaeilzadeh, Kamyar Azizzadenesheli, Karthik Kashinath, Mustafa Mustafa, Hamdi A. Tchelepi, Philip Marcus, Prabhat, and Anima Anandkumar. MeshfreeFlowNet: a physics-constrained deep continuous space-time super-resolution framework. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '20. IEEE Press, 2020. ISBN 9781728199986.

Kwang In Kim and Younghee Kwon. Single-image super-resolution using sparse regression and natural image prior. *IEEE transactions on pattern analysis and machine intelligence*, 32(6): 1127–1133, 2010.

Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 624–632, 2017.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations, 2021. URL `https://arxiv.org/abs/2010.08895`.

Ryo Okano and Masaaki Imaizumi. Distribution-on-distribution regression with wasserstein metric: Multivariate gaussian case. *Journal of Multivariate Analysis*, 202:105249, 2024.

Junier Oliva, Barnabas Poczos, and Jeff Schneider. Distribution to distribution regression. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 1049–1057, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR. URL `https://proceedings.mlr.press/v28/oliva13.html`.

Victor M. Panaretos and Yoav Zemel. *An Invitation to Statistics in Wasserstein Space*. Springer, 2020.

Alexander Petersen and Hans-Georg Müller. Fréchet regression for random objects with Euclidean predictors. *Annals of Statistics*, 47(2):691–719, 2019.

Gabriel Peyré and Marco Cuturi. *Computational Optimal Transport*. MIT Press, 2019.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. URL `https://arxiv.org/abs/1505.04597`.

Filippo Santambrogio. *Optimal Transport for Applied Mathematicians: Calculus of Variations, PDEs, and Modeling*. Birkhäuser, 2015.

Han Lin Shang and Rob J. Hyndman. Grouped functional time series forecasting: An application to age-specific mortality rates. *Journal of Computational and Graphical Statistics*, 26(2):330–343, 2017. doi: 10.1080/10618600.2016.1237877. URL `https://doi.org/10.1080/10618600.2016.1237877`.

Esteban G. Tabak, Giulio Trigila, and Wuchen Zhao. Data-driven conditional optimal transport. *Machine Learning*, 110:2705–2732, 2021.

Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer, 2009.

Cédric Villani. *Optimal Transport: Old and New*. Springer, 2009.

Junchi Wang et al. Neural conditional optimal transport. *arXiv preprint arXiv:2302.12345*, 2023.

Huixia Xu and Han Li. Wasserstein F-tests for Fréchet regression on Bures–Wasserstein manifolds. *Journal of Machine Learning Research*, 26:1–53, 2025.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan Salakhutdinov, and Alexander Smola. Deep sets, 2018. URL `https://arxiv.org/abs/1703.06114`.

A Technical Appendices and Supplementary Material

A.1 Data Generation

A.1.1 Gaussian Synthetic Data Generation

To evaluate the performance of our distribution-to-distribution regression framework, we first construct a synthetic Gaussian dataset in $\mathbb{R}^2$ with ground truth maps that are smoothly parameterized.

For each distribution pair (μ_i, ν_i), we generate data as follows:

1. Sample a mean vector $m_i \sim \text{Uniform}([-3, 3]^2)$.
2. Generate source samples $x_j^{(i)} \overset{\text{iid}}{\sim} \mathcal{N}(m_i, I_2)$, and define $\mu_i = \frac{1}{n}\sum_{j=1}^{n} \delta_{x_j^{(i)}}$.
3. Define a rotation angle $\theta_i = 2\|m_i\|$, and construct the corresponding rotation matrix:
$$A_i = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \end{bmatrix}.$$
4. Define a translation vector $a_i = 0.5 \cdot m_i$.
5. Define the ground truth affine map $T_i(x) = A_i x + a_i$.
6. Apply the map to each source sample and add Gaussian noise $\varepsilon_i \sim \mathcal{N}(0, \sigma^2 I_2)$ to produce the target samples:
$$y_j^{(i)} = T_i(x_j^{(i)}) + \varepsilon_i, \quad \nu_i = \frac{1}{n}\sum_{j=1}^{n} \delta_{y_j^{(i)}}.$$

The resulting dataset $\{(\mu_i, \nu_i)\}_{i=1}^{n}$ consists of empirical distributions related by smooth, locally varying affine maps. This setup reflects a structured but nontrivial regression problem while ensuring that the true map varies smoothly as a function of the source distribution.

A.1.2 Synthetic Gaussian Mixture Data Generation

We generate training pairs $\{(\mu_i, \nu_i)\}_{i=1}^n \subset \mathcal{P}_2(\mathbb{R}^2)$ where each μ_i is an empirical Gaussian mixture and ν_i is obtained by a piecewise smooth map applied to μ_i.

Fix $C \in \mathbb{Z}^+$ to be the number of mixture components, and $k \in \mathbb{Z}^+$ to be the number of samples per distribution. To generate the source distributions, μ_i, for each $i \in \{1, \dots, n\}$, we begin by sampling the mixture weights $w_i \sim \mathrm{Dirichlet}(\mathbf{1}_C)$. Then, we sample component means $m_{ij} \sim \mathrm{Unif}([L, U]^2))$ for $j = 1, \dots, C$, and draw component indices $c_\ell \sim \mathrm{Categorical}(w_i)$ for $\ell = 1, \dots, k$, sample points with isotropic covariance $\sigma^2 I_2$:

$$x_{i\ell} \sim \mathcal{N}\big(m_{i,c_\ell},\, \sigma^2 I_2\big), \qquad \ell = 1, \dots, k.$$

The empirical source distribution is

$$\mu_i = \frac{1}{k} \sum_{\ell=1}^{k} \delta_{x_{i\ell}}.$$

For each μ_i, define the barycenter $\bar{m}_i = \frac{1}{k} \sum_\ell x_{i\ell}$, and the RMS radius (equivalent to $W_2(\mu_i, \delta_0)$ in the continuum limit)

$$r_i \coloneqq \Big(\tfrac{1}{k} \sum_{\ell=1}^{k} \|x_{i\ell}\|^2\Big)^{1/2}.$$

Rotation angle $\theta(r) = \alpha r$, shear strength $k(r) = \kappa_0 + \kappa_1 r$. The matrices

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}, \qquad S(k) = \begin{bmatrix} 1 & k \\ 0 & 1 \end{bmatrix}$$

represent rotation and shear, respectively. Then, we define the operator,

$$A(r) = \lambda(r)\, R(\theta(r)) + (1 - \lambda(r))\, S(k(r)),$$

such that $\lambda(r) \in (0, 1)$ centered at r_{thresh} and

$$\lambda(r) = \sigma\Big(\tfrac{r_{\mathrm{thresh}} - r}{\gamma}\Big),$$

where $\gamma > 0$ controls the transition width. Then, with $a_i = \beta\, \bar{m}_i$, the samples are mapped by

$$y_{i\ell} = A(r_i)\, x_{i\ell} + a_i + \varepsilon_{i\ell}, \qquad \varepsilon_{i\ell} \sim \mathcal{N}(0,\, \tau^2 I_2),$$

forming the empirical target distribution

$$\nu_i = \frac{1}{K} \sum_{\ell=1}^{K} \delta_{y_{i\ell}}.$$