Active Learning of Conditional Generative Models via the Transport Neural Tangent Kernel

Jayoung Ryu[1] **Kyunghyun Cho**[1] **Romain Lopez**[1,2]

[1]Department of Computer Science, Courant Institute School of Mathematics, Computing, and Data Science, New York University
[2]Department of Biology, New York University

{jayoung.ryu, kyunghyun.cho, romain.lopez}@nyu.edu

Abstract

Many high-throughput scientific experiments produce distributional outputs, requiring conditional generative models to interpret the data, and to predict outcomes at new inputs. Acquiring experimental data at new conditions (e.g., for a new antigen or a new cell type) is often expensive, motivating the need for automated experimental design methods. However, existing active learning frameworks are limited as they assume scalar or vector-valued outputs. We first show that the Wasserstein test risk of a conditional generative task decomposes into a mean term and a shape term, demonstrating why mean-based active learning is blind to distributional shape. To capture both, we propose the *transport Neural Tangent Kernel* (tNTK), a tractable kernel that measures how sensitive a generative model's predicted distribution is with respect to its parameters. The tNTK recovers the standard NTK in the deterministic limit and upper-bounds the posterior Fréchet variance of the prediction under small parameter perturbations. It admits a closed form expression for affine Gaussian transport and tractable Monte Carlo estimators otherwise. Treating the tNTK as the covariance function of a Gaussian process, we greedily select inputs that minimize the total GP posterior variance across the pool, yielding a distribution-aware active learning strategy. We evaluate this strategy across four classes of conditional generative models, including conditional flow matching and conditional Monge gap, on synthetic distributions, intent-labeled sentences, and single-cell perturbation responses. In these experiments, the tNTK outperforms the baseline NTK computed from the conditional mean, demonstrating the value of distribution-aware active learning for generative-modeling tasks.

1 Introduction

Decades after a logic-based "robot scientist" first autonomously characterized gene functions in yeast metabolism [1, 2], machine learning has become a standard tool for the automated design of scientific experiments. These approaches have demonstrated impact across a wide range of scientific fields, including inorganic materials synthesis [3], drug discovery [4–7], and single-cell perturbation biology [8–12]. They have also been applied to large-scale image classification [13] and to the alignment of large language models [14, 15].

We consider the setup of pool-based active learning (AL) [16, 17], in which the acquisition strategy iteratively assembles a training set $\mathcal{X}_{\text{train}}$ from a set of candidate inputs $\mathcal{X}_{\text{pool}} \subseteq \mathcal{X}$. For each selected input $x \in \mathcal{X}_{\text{train}}$, the corresponding output $o \in \mathcal{O}$ is acquired using the experimental system. The goal of an AL strategy is to assemble a training set with a fixed number of data points so that a surrogate

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

predictive model $f(x)$ will generalize best to unseen inputs (i.e., a test set) — a problem extensively studied for classification [13, 18–22] ($\mathcal{O} = \{1, \ldots, K\}$) and regression [23–26] (single- or multi-task, $\mathcal{O} \subseteq \mathbb{R}^d$).

In many scientific applications, however, the experimental output is a probability distribution: $\mathcal{O} = \mathcal{P}(\mathcal{Y})$ for some underlying sample space $\mathcal{Y} \subseteq \mathbb{R}^d$, observed through a finite set of samples for each input. For example, chemical or genetic perturbation assays measure genomics data for hundreds of cells per perturbation, profiling entire distributions over cell states [8]. Regulatory genomics assays capture hundreds of thousands of hypersensitive DNA sequences in the genome for each cell type [27]. In both cases, the output mean $\mathbb{E}[y \mid x]$ is a poor substitute for the entire output $p(y \mid x)$ as the distribution itself is of scientific interest. For this reason, existing AL paradigms are not suited to distributional outputs because they have no other option but to reduce the output to a pre-specified scalar functional (e.g., mean or an event probability [28, 29]).

Conditional generative models such as conditional optimal transport [30, 31], flow matching [32], and diffusion models [33] are compelling options for predictive modeling of distributions. They demonstrated impressive performance for image and text generation [34, 35], as well as perturbation effect prediction [31, 36–39] and conditional DNA sequence generation [40]. Despite their adoption in these domains, no existing acquisition strategies directly leverage these generative models within an active learning framework for experimental design.

To address this gap, we develop a framework for active learning of conditional generative models that operates natively on distributional outputs. Specifically, we introduce the transport Neural Tangent Kernel (tNTK), a tractable kernel that measures the sensitivity of the predicted distribution $p_\theta(y \mid x)$ to perturbations of the model parameters θ. The tNTK admits two complementary interpretations: it upper-bounds the Fréchet posterior variance under a Laplace approximation to the parameter posterior, and it controls the metric derivative of the predicted distribution on the Wasserstein manifold. We characterize the geometric gap and show the bound is tight in dimension one. To propose the first distribution-aware active learning strategy, we treat the tNTK as a Gaussian Process (GP) covariance function, and greedily select inputs that minimize the total posterior variance across the pool, following existing kernel-based active learning frameworks [25, 26].

After introducing the necessary background (Section 2), we decompose the risk of a conditional generative prediction task into a mean and a shape term, and illustrate on a tractable 1D example that mean-based acquisition is suboptimal (Section 3). We then introduce the transport Neural Tangent Kernel (tNTK) as well as our distribution-aware AL strategy (Section 4). Finally, we conduct an empirical evaluation across four classes of conditional generative models — affine Gaussian transport, conditional quantile prediction, conditional flow matching, and conditional Monge gap — on simulated data where ground truth is available, as well as on real-world data from intent-labeled sentences and single-cell perturbation datasets (Section 5). Across 26 out of 30 settings, the tNTK outperforms baselines that operate on the conditional mean alone.

2 Background

2.1 Conditional Generative Models (CGMs)

We consider training data $\mathcal{D} = \{(x_i, \{y_{i,j}\}_{j=1}^{n_i})\}_{i=1}^{N}$, where each input $x_i \in \mathcal{X}_{\text{train}}$ is paired with a set of n_i independent and identically distributed samples $\{y_{i,j}\}_{j=1}^{n_i}$ drawn from an unknown conditional distribution $\mu(x_i) \in \mathcal{P}(\mathcal{Y})$. *Transport-based* CGMs induce output distributions by pushing forward a fixed base measure $\nu \in \mathcal{P}(\mathcal{Y})$ through a parametric map $T_\theta(x) : \mathcal{Y} \to \mathcal{Y}$:

$$\mu_\theta(x) = T_\theta(x)\#\nu, \tag{1}$$

where $\theta \in \mathbb{R}^p$ denotes the model parameters. Concretely, a sample from $\mu_\theta(x)$ is obtained by drawing $y \sim \nu$ and returning $T_\theta(x)(y)$. This formulation covers various common families of conditional generative models, two of which we now briefly detail.

Conditional optimal transport Conditional optimal transport frameworks learn a parametric map $T_\theta(x)$ that approximates the optimal transport map from ν to $\mu(x)$ across all input points $x \in \mathcal{X}_{\text{train}}$. CondOT [30] parameterizes $T_\theta(x)(y) = \nabla_y \Phi_\theta(x, y)$ where Φ_θ is a partially input convex neural network (PICNN) [41], which ensures that $T_\theta(x)$ is a valid Monge map. Conditional Monge gap [31]

drops the PICNN constraint and instead trains an unconstrained neural network with two terms: a distributional loss between $T_\theta(x)\#\nu$ and $\mu(x)$, and the Monge gap regularizer [42], which penalizes the suboptimality of $T_\theta(x)$ as a transport map between ν and its own pushforward. This approach improves numerical stability and empirical performance over CondOT.

Conditional flow matching (CFM) CFM [32][1] learns a parametric, time- and condition-dependent vector field $v_\theta^{t,x} : \mathcal{Y} \to \mathcal{Y}$ by regressing on a target field $u^{t,x}$ constructed from a chosen conditional probability path interpolating between ν at $t = 0$ and $\mu(x)$ at $t = 1$. The flow $\phi^{t,x}(y)$ is defined as the solution of the following ordinary differential equation:

$$\frac{d}{dt}\phi^{t,x}(y) = v_\theta^{t,x}(\phi^{t,x}(y)), \quad \phi^{0,x}(y) = y. \tag{2}$$

At inference time, samples from $\mu_\theta(x) = (\phi^{1,x})\#\nu$ are obtained by sampling $y \sim \nu$ and integrating the ODE from $t = 0$ to $t = 1$ with initial condition $\phi^{0,x}(y) = y$. Although CFM specifically applies to the case where ν is a noise distribution, variants such as OT-CFM [43] extend CFM to cases where ν is a data distribution using optimal transport couplings.

2.2 Active Learning for Regression

In pool-based AL, the acquisition strategy has access to a candidate set $\mathcal{X}_{\text{pool}} \subseteq \mathcal{X}$ of unlabeled inputs and selects a subset $\mathcal{S} \subseteq \mathcal{X}_{\text{pool}}$ for which outputs are measured with the experimental system. A key heuristic in the field is to select the input that minimizes the total posterior variance of the surrogate model f_θ on the pool,

$$x^\star = \arg\min_{x' \in \mathcal{X}_{\text{pool}}} \sum_{x \in \mathcal{X}_{\text{pool}}} \text{Var}[f_\theta(x) \mid \mathcal{X}_{\text{train}} \cup \{x'\}]\,. \tag{3}$$

Such candidates are often the most informative for generalization to a test set [26]. Active learning strategies for regression are further discussed in [25].

For neural network-based models, predictive uncertainty is intractable in general and must be approximated. Common approaches include approximate Bayesian inference over the model parameters [44], Monte-Carlo dropout [45], and kernel-based approximations derived from the finite-width Neural Tangent Kernel (NTK) [46, 47]:

$$k_\theta^{\text{NTK}}(x, x') = \nabla_\theta f(x)^\top \nabla_\theta f(x'). \tag{4}$$

The predictive distribution f after linearization around the trained parameters θ is well approximated by a Gaussian process with covariance k_θ^{NTK}. Informative candidates can then be selected using standard kernel-based acquisition strategies [25], such as the total-posterior-variance minimization described above. All of these methods assume that $f(x)$ is a scalar or low-dimensional prediction, and do not extend to settings where the model's output is a probability distribution.

3 Distributional Risk Decomposition

We illustrate the need for distribution-aware AL methods by analyzing the test risk of conditional generative models under the 2-Wasserstein loss. These derivations suggest that existing AL methods based on test risk minimization for conditional mean prediction are suboptimal. We then present a fully-tractable one-dimensional example alongside a numerical experiment.

3.1 Mean-shape decomposition of distributional risk

For the following theoretical derivations, we assume that distribution $\mu(x_i)$ (or equivalently, infinite samples from it) is observed for each input x_i. This allows us to define our training data as $\mathcal{D} = \{(x_i, \mu(x_i))\}_{i=1}^N$. We also assume the availability of a posterior distribution $\theta \mid \mathcal{D}$ over the parameters

[1] We use "conditional" to refer to conditioning on an external covariate x (e.g., [37]). This differs from the original usage in Lipman et al. [32], where "conditional" refers to per-sample conditional probability paths used to train an unconditional vector field.

θ of a CGM that reflects uncertainty in the predicted distribution $\mu_\theta(x) = T_\theta(x)_{\#}\nu$. For a test point $x \in \mathcal{X}$, the risk under the squared Wasserstein loss is defined as:

$$\mathcal{R}(x, \mathcal{D}) = \mathbb{E}_{\theta|\mathcal{D}}\big[W_2^2(\mu_\theta(x), \mu(x))\big], \tag{5}$$

and admits the following decomposition:

Proposition 1 (Mean-shape decomposition of risk)**.** *The Wasserstein risk defined in Equation (5) decomposes as*

$$\mathcal{R}(x, \mathcal{D}) = \underbrace{\mathbb{E}_{\theta|\mathcal{D}}\big[\|m_\theta(x) - m(x)\|^2\big]}_{\mathcal{R}_{mean}(x,\mathcal{D})} + \underbrace{\mathbb{E}_{\theta|\mathcal{D}}\big[W_2^2(\mu_\theta^c(x), \mu^c(x))\big]}_{\mathcal{R}_{shape}(x,\mathcal{D})}, \tag{6}$$

where $m_\rho = \int y \, d\rho(y)$ denotes the mean of ρ, and ρ^c is obtained by centering ρ for any $\rho \in \mathcal{P}_2(\mathcal{Y})$.

The proof is given in Appendix A.1. Classical regression-based active learning traditionally aims at (approximately) minimizing the mean-risk $\mathbb{E}_{x\sim p_{\text{test}}}\mathcal{R}_{\text{mean}}(x, \mathcal{D})$ [23]. Because they do not consider the additional shape-related term, these methods may perform poorly whenever shape matters.

3.2 One-dimensional example

The one-dimensional case is a remarkable instance since Wasserstein distances admit closed-form expressions. Let $F^{-1}(x;\tau)$ (resp. $F_\theta^{-1}(x;\tau)$) denote the τ-th quantile of distribution $\mu(x)$ (resp. $\mu_\theta(x)$) for $\tau \in [0,1]$. We have the following identities:

Lemma 1 (1D posterior risk)**.** *Assume the model is unbiased at the mean and every quantile level, i.e., $\mathbb{E}_{\theta|\mathcal{D}}[m_\theta(x)] = m(x)$ and $\mathbb{E}_{\theta|\mathcal{D}}[F_\theta^{-1}(x;\tau)] = F^{-1}(x;\tau)$ for all x, τ. Then*

$$\mathcal{R}_{mean}(x, \mathcal{D}) = \mathrm{Var}_{\theta|\mathcal{D}}[m_\theta(x)], \qquad \mathcal{R}_{shape}(x, \mathcal{D}) = \int_0^1 \mathrm{Var}_{\theta|\mathcal{D}}[g_\theta(x;\tau)]\, d\tau, \tag{7}$$

where $g_\theta(x;\tau) = F_\theta^{-1}(x;\tau) - m_\theta(x)$ denotes the centered quantile function of $\mu_\theta(x)$.

The proof appears in Appendix A.2. The shape-related term becomes tractable as it reduces to an integral of marginal posterior variances over quantile levels $\tau \in [0,1]$.

We numerically illustrate this on simulated data (details in Appendix D.1). Briefly, $\mu(x)$ is a two-component Gaussian mixture model with mean and variance depending on $x \in [0,1]$ (Figure 1A). $m(x)$ and $g(x;\tau)$ are modeled for a discrete grid of τ levels using independent Gaussian processes. This approach provides direct access to the posterior variances in (7). We run AL campaigns to compare three acquisition strategies: Wasserstein AL, which minimizes the expected total risk, mean AL, which minimizes only its mean component,

$$x^*_{\text{W}} = \arg\min_{x' \in \mathcal{X}_{\text{pool}}} \mathbb{E}_{x\sim p_{\text{test}}}\mathcal{R}(x, \mathcal{D}_{\text{train}} \cup \{x'\}), \tag{8}$$

$$x^*_{\text{mean}} = \arg\min_{x' \in \mathcal{X}_{\text{pool}}} \mathbb{E}_{x\sim p_{\text{test}}}\mathcal{R}_{\text{mean}}(x, \mathcal{D}_{\text{train}} \cup \{x'\}), \tag{9}$$

and random sampling.

Wasserstein AL outperforms Mean AL by over an order of magnitude (Figure 1B). Mean AL still improves over random in terms of mean error (Figure 1C) but plateaus on shape error (Figure 1D). These results illustrate that ignoring distribution shape for active learning is suboptimal.

4 Transport Kernels for Active Learning of Conditional Distributions

For most applications of interest, neither the test risk nor the shape term from (6) is tractable. Most fundamentally, the ground-truth distribution $\mu(x)$ is unknown at unseen inputs $\mathcal{X}_{\text{pool}}$. Even if it were known, the 2-Wasserstein distance remains intractable for neural transport maps. Those represent significant challenges for distribution-aware active learning.

We circumvent these issues by considering the posterior Fréchet variance as a measure of uncertainty in a generative model's predictions $\mu_\theta(x)$. We bound this quantity by linearizing the transport map $T_\theta(x)$ around its parameters θ, which yields the transport Neural Tangent Kernel (tNTK). We briefly discuss properties of the tNTK, how to estimate it efficiently, and then use it to design a distribution-aware active learning strategy.

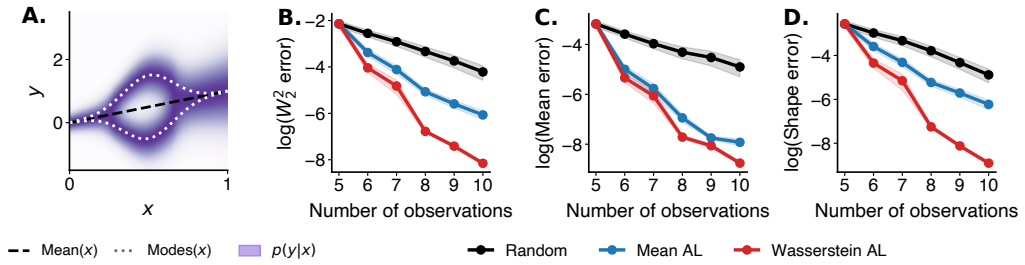

Figure 1: Synthetic conditional density estimation. (A) Ground truth conditional density $\mu(x)$. (B–D) Test error after active learning as a function of training set size: total risk (B), mean prediction error (C), and shape error (D). Results averaged over 50 random initializations; shaded bands show ±2 standard errors.

4.1 Quantifying model uncertainty via parameter sensitivity

We consider the *posterior Fréchet variance*,

$$V_F(x) := \mathrm{Var}_{W_2}[\mu_\theta(x) \mid \mathcal{D}] = \inf_{\rho \in \mathcal{P}_2(\mathcal{Y})} \mathbb{E}_{\theta|\mathcal{D}}\big[W_2^2(\mu_\theta(x), \rho)\big], \tag{10}$$

defined as the minimum mean squared Wasserstein distance from a fixed reference measure to a posterior draw $\mu_\theta(x)$ [48]. The minimizer is the Wasserstein barycenter $\bar{\mu}_\theta(x)$ of the posterior law of $\mu_\theta(x)$ [49]. The posterior Fréchet variance $V_F(x)$ has two properties that make it especially relevant for active learning. First, it lower bounds the risk $V_F(x) \le \mathcal{R}(x, \mathcal{D})$, with equality when the Wasserstein barycenter equals the data-generating distribution $\bar{\mu}_\theta(x) = \mu(x)$. Second, in the deterministic limit, where $\mu_\theta(x)$ collapses to a Dirac mass at $f_\theta(x)$, $V_F(x)$ recovers the standard posterior variance $\mathrm{Var}_{\theta|\mathcal{D}}[f_\theta(x)]$.

Although $V_F(x)$ is intractable in general, since the Wasserstein barycenter of the posterior law of $\mu_\theta(x)$ has no closed form, we show that linearizing the transport map yields a tractable upper bound.

Theorem 1 (Upper bound on the posterior Fréchet variance)**.** *Let $\theta \mid \mathcal{D} \sim \mathcal{N}(\hat{\theta}, \Sigma)$, with $\Sigma \in \mathbb{R}^{p\times p}$ positive definite and $\|\Sigma\|_{\mathrm{op}} < 1$. Assume that for every $x \in \mathcal{X}$ and for ν-almost every $y \in \mathcal{Y}$, the map $\theta \mapsto T_\theta(x)(y)$ is $C^2(\mathbb{R}^p)$, and that the Jacobian $D_\theta T_{\hat{\theta}}(x)(\cdot)$ and the pointwise supremum of the operator norm of the Hessian, $\sup_\theta \|D_\theta^2 T_\theta(x)(\cdot)\|_{\mathrm{op}}$, are in $L^2(\nu)$. Then, there exists $C(x)$ such that:*

$$V_F(x) \le \mathrm{Tr}\big(G_{\hat{\theta}}(x)\,\Sigma\big) + C(x)\|\Sigma\|_{\mathrm{op}}^{3/2}, \tag{11}$$

where the $p \times p$ parameter sensitivity matrix at input x is

$$G_{\hat{\theta}}(x) = \int D_\theta T_{\hat{\theta}}(x)(y)^\top D_\theta T_{\hat{\theta}}(x)(y)\, d\nu(y). \tag{12}$$

Inequality (11) is tight to leading order when $\dim(\mathcal{Y}) = 1$ and $T_\theta(x)$ is monotone in y.

The proof is given in Appendix A.3. Theorem 1 identifies that under a Laplace approximation [50, 51] to the parameter posterior, $\mathrm{Tr}\big(G_{\hat{\theta}}(x)\,\Sigma\big)$ plays the role of a tractable measure of the predictive variance in the generative model's predictions, in the limit of small variance. This is important given that exact computations of the Wasserstein metric tensor are otherwise intractable for neural transport maps [52, 53].

The tightness of the bound from (11) is an important consideration that we investigate in several ways. First, we give a geometric interpretation of $G_\theta(x)$ by upper-bounding the Wasserstein metric derivative of trajectories of $\mu_\theta(x)$ for small deviations around θ and characterize when the bound is tight (Appendix B). Second, for affine transport of Gaussian distributions (Appendix E.1), we find numerically that the bound is exact for $d = 1$ and overestimates the true variance by at most 70% for $d = 50$, i.e., by less than a factor of two. Finally, our experiments on active learning benchmarks (Section 5) suggest that the parameter sensitivity matrix $G_{\hat{\theta}}(x)$ is informative for active learning across multiple model classes.

4.2 Transport Neural Tangent Kernel

The parameter sensitivity matrix $G_\theta(x)$ extends naturally to a pairwise object by considering two distinct inputs x and x'. Specifically, we define the *transport Neural Tangent Kernel* (tNTK) as:

$$k_\theta(x, x') = \int \mathrm{Tr}\left[D_\theta T_\theta(x)(y)^\top D_\theta T_\theta(x')(y)\right] d\nu(y). \tag{13}$$

This kernel measures the alignment between the parameter sensitivities of the transport maps at x and x', integrated over the base measure ν. The diagonal of the kernel recovers the trace of the parameter sensitivity matrix, $k_\theta(x, x) = \mathrm{Tr}[G_\theta(x)]$. Under an isotropic posterior $\Sigma = \sigma^2 I$ – the standard choice in kernel-based active learning [25] – Theorem 1 implies that

$$V_F(x) \le \sigma^2 k_\theta(x, x) + O(\sigma^3). \tag{14}$$

This identifies the kernel diagonal as a tractable scalar measure of predictive uncertainty. The tNTK is positive semi-definite (Appendix A.4). When ν degenerates to a Dirac measure, $\mu_\theta(x)$ collapses to a single point and the tNTK reduces to the finite-width NTK [46], justifying the name.

Kernel Computation and Estimation The tNTK may be computed over all parameters θ or restricted to the last layer parameters. When the transport map takes the form $T_\theta(x)(y) = W^\top h_\theta(x, y)$ with penultimate features $h_\theta(x, y) \in \mathbb{R}^m$ and a linear last layer $W \in \mathbb{R}^{m\times d}$, the tNTK has a simple expression:

$$k_\theta^{\mathrm{ll}}(x, x') = d \int h_\theta(x, y)^\top h_\theta(x', y)\, d\nu(y). \tag{15}$$

The proof appears in Appendix A.5. (15) involves the inner product of penultimate features at x and x' integrated over the base measure ν and therefore represents the distributional analog of the last-layer NTK [25]. In practice, we approximate the integral with Monte Carlo samples from ν, requiring only forward passes through the network h_θ.

4.3 Kernel Decomposition for Affine Gaussian Transport Maps

We show that when $T_\theta(x)$ is an affine transport map, the tNTK admits a clean decomposition into mean- and shape-sensitivity terms.

Lemma 2. *Let us assume that $T_\theta(x)$ is an affine map: $T_\theta(x)(y) = b_\theta(x) + M_\theta(x)y$ and that the base measure ν is centered and isotropic (i.e., $\mathbb{E}_{y\sim\nu}[y] = 0$ and $\mathbb{E}_{y\sim\nu}[yy^\top] = I$). Then*

$$k_\theta(x, x') = \mathrm{Tr}\left[D_\theta b_\theta(x)^\top D_\theta b_\theta(x')\right] + \sum_{j=1}^{d} \mathrm{Tr}\left[D_\theta M_\theta^j(x)^\top D_\theta M_\theta^j(x')\right], \tag{16}$$

where $M_\theta^j(x)$ denotes the j-th column of $M_\theta(x)$.

The proof appears in Appendix A.6. This decomposition mirrors the one from Proposition 1 at the kernel level: the first term captures sensitivity of the predicted mean, the second captures sensitivity of the predicted shape. Additionally, we demonstrate that when $M_\theta(x)$ is further restricted to be lower-triangular (e.g., via Cholesky decomposition), for any perturbation $h \in \mathbb{R}^p$, $h^\top G_\theta(x)h$ coincides with the squared metric derivative at 0 of the curve $t \mapsto \mu_{\theta+th}(x)$ under the Knothe–Rosenblatt (KR) transport metric (Appendix C). This suggests that in the setting of linear maps, the parameter sensitivity matrix $G_\theta(x)$ effectively captures shape variations.

4.4 Active Learning Methods via the Transport NTK

Let $\mathcal{X}_{\text{train}}$ denote the conditions for which we have observed output samples, and $\mathcal{X}_{\text{pool}}$ the set of inputs for which experimental data is not available. Treating the tNTK as the covariance function of a Gaussian process, the GP posterior variance at a test point x' after conditioning on a set $\mathcal{X}_S \subseteq \mathcal{X}$ is:

$$\sigma_\theta^2(x' \mid \mathcal{X}_S) = k_\theta(x', x') - k_\theta(\mathcal{X}_S, x')^\top \left(k_\theta(\mathcal{X}_S, \mathcal{X}_S) + \lambda I\right)^{-1} k_\theta(\mathcal{X}_S, x'), \tag{17}$$

where $\lambda > 0$ is a regularization parameter. We greedily select the next candidate as the one that minimizes the average posterior variance over the whole dataset:

$$x^* = \underset{x\in\mathcal{X}_{\text{pool}}}{\arg\min} \; \frac{1}{|\mathcal{X}_{\text{pool}}| + |\mathcal{X}_{\text{train}}|} \sum_{x'\in\mathcal{X}_{\text{pool}}\cup\mathcal{X}_{\text{train}}} \sigma_\theta^2(x' \mid \mathcal{X}_{\text{train}} \cup \{x\}). \tag{18}$$

We apply this rule sequentially to design an entire batch (Algorithm 1).

Algorithm 1 Active learning with the transport neural tangent kernel

Require: Pool of conditions $\mathcal{X}$, batch size B, number of rounds T, regularization $\lambda > 0$

Initialize $\mathcal{X}_{\text{train}} \subset \mathcal{X}$ with $|\mathcal{X}_{\text{train}}| = B_0$; set $\mathcal{X}_{\text{pool}} = \mathcal{X} \setminus \mathcal{X}_{\text{train}}$
for $t = 1, \ldots, T$ **do**
  Fit generative model on $\mathcal{X}_{\text{train}}$ to obtain parameters θ_t
  $\mathcal{B} \leftarrow \varnothing$ ▷ current batch
  $\mathcal{X}_{\mathcal{C}} \leftarrow \mathcal{X}_{\text{train}}$ ▷ conditioning set
  for $b = 1, 2, \ldots, B$ **do**
    $x^* \leftarrow \underset{x \in \mathcal{X}_{\text{pool}} \setminus \mathcal{B}}{\arg\min} \sum_{x' \in \mathcal{X}} \sigma^2_{\theta_t}(x' \mid \mathcal{X}_{\mathcal{C}} \cup \{x\})$ ▷ select point
    $\mathcal{B} \leftarrow \mathcal{B} \cup \{x^*\}$
    $\mathcal{X}_{\mathcal{C}} \leftarrow \mathcal{X}_{\mathcal{C}} \cup \{x^*\}$
  $\mathcal{X}_{\text{train}} \leftarrow \mathcal{X}_{\text{train}} \cup \mathcal{B}$ ▷ update training set
  $\mathcal{X}_{\text{pool}} \leftarrow \mathcal{X}_{\text{pool}} \setminus \mathcal{B}$ ▷ update pool
return Final model parameters θ_T

5 Experiments

We investigate the empirical performance of our active learning strategy based on the tNTK, following Algorithm 1, across four conditional generative modeling tasks. We consider as baseline the selection rule in (18) applied to the NTK computed on a neural network fitting the mean of the distribution ("NTK") and a random sampling baseline (see Appendix D.2 for more details). The NTK baseline matches BAIT [26] as defined in Holzmüller et al. [25], and is among the best active learning methods for regression tasks in Holzmüller et al. [25]. Unless mentioned otherwise, we use the Monte Carlo estimate of the last layer tNTK for all experiments with finite data (using $M = 4,096$ samples). Full experimental details can be found in Appendix D.

5.1 Affine transport of 2D Gaussian distributions

We first investigate the active learning performance of the kernel when $G_\theta(x)$ can be computed in closed form (Lemma 2). We consider 2-dimensional isotropic Gaussian $y \sim \mathcal{N}(0, I_2)$ and conditional transport maps of the form

$$T_\theta(x)(y) \;=\; b_\theta(x) \;+\; L_\theta(x) y\,, \tag{19}$$

where $b_\theta(x) \in \mathbb{R}^2$ is the predicted mean and $L_\theta(x) \in \mathbb{R}^{2\times 2}$ is a lower-triangular matrix with strictly positive diagonal entries, so that $\Sigma_\theta(x) = L_\theta(x)\, L_\theta(x)^\top$ is a valid covariance matrix. $b_\theta(x)$ and $L_\theta(x)$ are parameterized by neural networks, and trained against the analytical Gaussian parameters observed for each input x (under an infinite-sample assumption). We generate data so that the input region $x_1 \approx 0$ has high variation in the mean but nearly constant covariance, with the opposite pattern for the region $x_1 \approx 1$ (Figure 2A-B). While the mean-based NTK does not improve over random

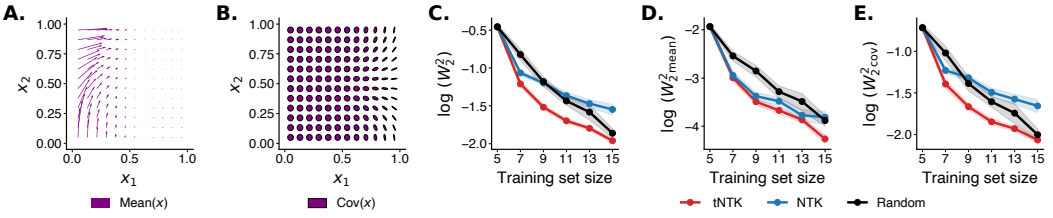

Figure 2: Results on the synthetic Gaussian 2D dataset. (A) Ground truth conditional mean $\mathbb{E}[y \mid x]$. (B) Ground truth conditional covariance $\mathrm{Cov}[y \mid x]$. (C) Test Wasserstein error after active learning as a function of training set size (lower is better). (D-E) Mean (D) and covariance (E) components of the test Wasserstein error. Results are averaged over 30 independent random initializations; shaded bands show standard errors.

Table 1: Change in Wasserstein test risk from iteration 0 (lower is better). For each seed, we average over $T = 5$ iterations and $K = 5$ folds; we then report mean ± standard error across $N = 10$ seeds, in units of 10^{-3}. Column headers indicate $B_0 + B$, the initial training set size and the number of conditions added per AL round. Best mean per column shown in bold.

Dataset	Strategy	5 + 1	5 + 5	10 + 1
Gaussian mixture	Random	-296.12 ± 16.29	-609.61 ± 17.10	-121.02 ± 10.93
	NTK	-416.43 ± 16.15	-685.12 ± 17.58	-168.28 ± 11.23
	NTK (mean-var)	-367.99 ± 17.59	-686.29 ± 17.39	-162.85 ± 11.04
	NTK (quartile)	-388.93 ± 16.43	-681.29 ± 17.62	-176.86 ± 10.72
	BALSA-EMD	-332.16 ± 16.06	-552.52 ± 17.07	-133.59 ± 12.12
	tNTK	$\mathbf{-429.52 \pm 16.39}$	$\mathbf{-695.82 \pm 17.89}$	$\mathbf{-205.01 \pm 12.38}$
Temperature	Random	-89.04 ± 12.88	-219.96 ± 14.43	-41.02 ± 7.44
	NTK	-185.55 ± 14.66	-255.65 ± 15.29	-84.57 ± 9.70
	NTK (mean-var)	-186.51 ± 15.31	-258.48 ± 15.55	-86.88 ± 9.69
	NTK (quartile)	-182.63 ± 14.97	-255.01 ± 15.86	-88.36 ± 9.66
	BALSA-EMD	-155.37 ± 13.64	-199.14 ± 15.87	-71.14 ± 9.57
	tNTK	$\mathbf{-189.94 \pm 15.22}$	$\mathbf{-267.93 \pm 15.36}$	$\mathbf{-89.84 \pm 10.01}$
Dew Point	Random	-79.90 ± 14.55	-280.77 ± 21.75	-76.55 ± 10.15
	NTK	-178.44 ± 18.98	-364.19 ± 21.53	-158.94 ± 10.97
	NTK (mean-var)	-172.47 ± 17.97	-372.49 ± 22.06	-159.24 ± 10.32
	NTK (quartile)	-173.21 ± 18.59	-375.47 ± 22.57	-158.70 ± 11.14
	BALSA-EMD	-79.51 ± 14.23	-178.00 ± 17.91	-124.13 ± 9.39
	tNTK	$\mathbf{-221.18 \pm 17.13}$	$\mathbf{-442.25 \pm 22.34}$	$\mathbf{-212.74 \pm 11.73}$

selection, the tNTK achieves a 1.2x lower Wasserstein loss than random by the end of the campaign (Figure 2C). As in the one-dimensional example of Section 3.2, NTK improves over random in mean loss but not in covariance loss (Figure 2D-E). We further show that this result also holds for the affine transport of d-dimensional Gaussian distributions, where the tNTK outperforms NTK up to $d = 50$ (Appendix E.2). Model and experimental details appear in Appendix D.3.

5.2 Conditional quantile prediction

We next consider the case where $\dim(\mathcal{Y}) = 1$ with $T_\theta(x)$ monotone in y. As shown in Appendix A.3, this yields the quantile representation $T_\theta(x) = F_\theta^{-1}(x;\cdot) \circ F_\nu$, where $F_\theta^{-1}(x;\cdot)$ is the quantile function of $\mu_\theta(x)$. In this setting, the bound from Theorem 1 is tight at the leading order, and the tNTK admits the following closed-form expression:

$$k_\theta^{\mathrm{1D}}(x, x') = \int_0^1 \nabla_\theta F_\theta^{-1}(x;\tau)^\top \nabla_\theta F_\theta^{-1}(x';\tau)\, d\tau. \tag{20}$$

We discretize τ on a uniform grid and parameterize conditional quantiles $F_\theta^{-1}(x;\tau)$ with non-crossing quantile networks [54]. We evaluate the performance of our method on one simulated and two real datasets. Simulated data are based on three-component Gaussian mixture models whose parameters are functions of the input x. Real data are based on yearly distributions of surface temperature and dew point [55], predicted from the location of the measuring station.

In this experiment, we include as baseline the NTK from multiple networks that were fitted on summary statistics of the data, such as the mean, the mean and the variance (mean-var), as well as all quartiles (quartile) of the distribution. We also report results for Bayesian Active Learning By Distribution Disagreement (BALSA^{EMD}) [56].

The tNTK consistently outperforms all baselines across differing numbers of initial conditions (B_0) and the number of conditions selected in each iteration (B) (Table 1 and Figure S2). Model and dataset details appear in Appendix D.4.

5.3 Conditional Flow Matching

In conditional flow matching, the transport map is the time-1 flow $T_\theta(x)(y) = \phi^{1,x}(y)$ obtained by integrating the ODE from Equation 2. We approximate the total derivative by the integration of

Table 2: Change in Sinkhorn divergence from iteration 0 (lower is better). For each seed, we average over $T = 3$ iterations and $K = 5$ folds; we then report mean ± standard error across $N = 10$ seeds, in units of 10^{-3}. Column headers indicate $B_0 + B$, the initial training set size and the number of conditions added per AL round. Best mean per column shown in bold.

Dataset	Strategy	5 + 1	5 + 5	10 + 5
Radial flow	Random	−17.70 ± 3.97	−40.24 ± 8.42	−24.60 ± 4.16
	NTK	−26.71 ± 6.35	−43.07 ± 7.61	−24.24 ± 4.86
	tNTK	**−35.50 ± 5.36**	**−53.55 ± 6.83**	**−38.39 ± 5.03**
Van der Pol oscillator	Random	−121.87 ± 31.66	−336.33 ± 34.17	−171.39 ± 28.31
	NTK	−196.36 ± 32.59	−408.15 ± 37.03	−220.98 ± 33.08
	tNTK	**−200.73 ± 37.15**	**−441.51 ± 24.25**	**−259.98 ± 37.37**
CLINC 150	Random	−350.07 ± 91.54	−883.23 ± 155.74	−436.98 ± 133.64
	NTK	−306.01 ± 111.84	**−884.02 ± 162.84**	−480.66 ± 96.75
	tNTK	**−672.85 ± 107.66**	−852.31 ± 133.11	**−580.94 ± 93.76**

$D_\theta v_\theta^{t,x}$, and compute the tNTK as

$$k_\theta^{\text{Flow}}(x, x') = \int \text{Tr}\left[\left(\int_0^1 D_\theta v_\theta^{t,x}(\phi^{t,x}(y))\,dt\right)^\top \left(\int_0^1 D_\theta v_\theta^{t,x'}(\phi^{t,x'}(y))\,dt\right)\right] d\nu(y). \tag{21}$$

We estimate the integral over ν in (21) with Monte Carlo sampling, and compute both the inner time integration and the flow ODE for $\phi^{t,x}(y)$ with Tsitouras' 5/4 adaptive Runge–Kutta method [57]. For this experiment, we use a Gaussian conditional probability path with mean $\mu_t = t\,y_1$ and standard deviation $\sigma_t = 1 - t$, where $y_1 \sim \mu(x)$ is a target sample and the path interpolates between $\mathcal{N}(0, I)$ at $t = 0$ and a Dirac at y_1 at $t = 1$, following Lipman et al. [32].

We evaluate our kernel on two simulated and one real-world dataset. The simulated systems generate particle distributions in $\mathbb{R}^2$: radial flows with attractors at varying radii and steepness, and Van der Pol oscillators [58] with varying parameters and initial particle distributions; we use 100 conditions for each. For the real-world data, we use the CLINC150 intent classification dataset [59, 60]: each conditioning input is the fine-tuned MINILM [61] text embedding of an intent class label, and the target distribution is the empirical distribution of sentences in that class, embedded into an 8-dimensional latent space using PCA on BART [62] sentence embeddings.

tNTK outperforms NTK across 8 out of 9 (dataset, budget) configurations (Table 2 and Figure S3). In the remaining configuration (CLINC 150; $B_0 = 5$; $B = 5$), NTK does not significantly outperform tNTK ($p > 0.5$, paired t-test). This result demonstrates that the kernel remains useful even when its computation requires numerical integration of the flow ODE. Full experimental details appear in Appendix D.5.

5.4 Conditional Monge Gap

We use the conditional Monge gap [31, 42] to predict single-cell responses in two perturbation datasets, where mapping unperturbed cell states to perturbed cell states has often been modeled using optimal transport [63, 36]. The K562-Chemical task uses the SciPlex3 dataset [64]: x is the mean transcriptomic response in MCF7 cells (adenocarcinoma cell line) under a small-molecule drug treatment, and the target is the single-cell transcriptomic response in K562 cells (leukemia cell line) under the same treatment, embedded into a 10-dimensional latent space. The BMDM-Genetic task uses a genetic screen of bone marrow-derived macrophages: x is the imaging outcome from a matched optical pooled screen, and the target is the transcriptomic response in a 10-dimensional latent space. Data processing details appear in Appendix D.6.

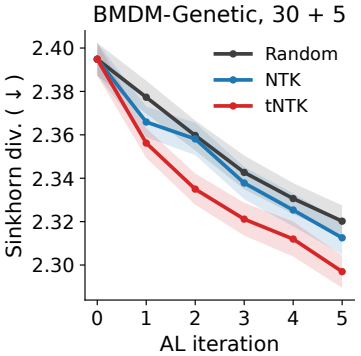

Figure 3: Mean Sinkhorn divergence per active learning iteration across 5 folds and 10 random seeds. Shade shows standard error.

Table 3: Change in Sinkhorn divergence from iteration 0 (lower is better). For each seed, we average over $T = 5$ iterations and $K = 5$ folds; we then report mean ± standard error across $N = 10$ seeds, in units of 10^{-3}. Column headers indicate $B_0 + B$, the initial training set size and the number of conditions added per AL round. Best mean per column shown in bold.

Dataset	Strategy	$B_0 + B$		
		10 + 1	10 + 5	20 + 5
K562-Chemical	Random	$\mathbf{-9.56 \pm 0.69}$	-29.06 ± 1.09	-15.27 ± 0.69
	NTK	-9.33 ± 0.75	$\mathbf{-30.44 \pm 1.45}$	-20.04 ± 0.80
	tNTK	-8.65 ± 1.16	-30.26 ± 1.48	$\mathbf{-21.00 \pm 1.30}$
		20 + 10	30 + 5	30 + 10
	Random	-24.14 ± 1.29	-10.70 ± 1.19	-17.19 ± 1.34
	NTK	-30.86 ± 0.84	-17.57 ± 1.06	-25.53 ± 0.97
	tNTK	$\mathbf{-31.63 \pm 1.22}$	$\mathbf{-19.30 \pm 1.18}$	$\mathbf{-26.13 \pm 1.13}$
		10 + 1	10 + 5	20 + 5
BMDM-Genetic	Random	-23.67 ± 5.56	-98.81 ± 6.38	-67.70 ± 5.29
	NTK	$\mathbf{-38.16 \pm 4.98}$	-127.30 ± 6.24	-92.80 ± 5.00
	tNTK	-32.01 ± 4.33	$\mathbf{-130.46 \pm 7.60}$	$\mathbf{-102.63 \pm 5.20}$
		20 + 10	30 + 5	30 + 10
	Random	-111.13 ± 3.87	-48.73 ± 2.41	-79.70 ± 3.97
	NTK	-130.32 ± 5.70	-54.90 ± 4.17	-90.86 ± 4.54
	tNTK	$\mathbf{-144.90 \pm 3.50}$	$\mathbf{-70.59 \pm 4.84}$	$\mathbf{-106.25 \pm 4.91}$

tNTK outperforms NTK and random baselines in 9 out of 12 (dataset, budget) configurations (Table 3 and Figure S4). In the remaining cases, NTK did not outperform significantly the tNTK ($p > 0.1$, paired t-test).

6 Discussion

We introduced an active learning framework for conditional distributions based on the *transport neural tangent kernel*. The tNTK captures the sensitivity of a conditional generative model's predicted distribution to perturbations of its parameters, and generalizes the NTK to distributional outputs. We derived the tNTK as an upper bound on the posterior Fréchet variance (Theorem 1) and provided a geometric interpretation on the Wasserstein manifold (Appendix B). Across four conditional generative model classes and on simulated and real datasets, total-posterior-variance-minimizing selection with the tNTK as GP covariance function outperforms the same strategy applied to the NTK of the conditional mean, as well as random sampling.

Sequential greedy total posterior variance minimization, while a principled choice, may be outperformed by selection methods explicitly designed for batch diversity [25], especially in large-batch regimes. However, the tNTK can be paired with any kernel-based selection method, so combining it with existing or novel selection strategies offers a natural extension of our framework to large-batch settings.

The framework extends naturally to other model classes, including diffusion models: in their probability-flow ODE formulation [65], the model is deterministic and the same Jacobian-integration argument as in CFM applies. Conditional normalizing flows [66] provide direct access to the inverse map and pushforward density, and score-based kernels with theoretical guarantees in the regression setting [67, 19, 25] could be extended to the distributional case as an alternative to the tNTK.

In dimensions greater than one, neither the Fréchet variance bound (Theorem 1) nor the metric derivative bound (Lemma 4) is generally tight. While we have characterized the gap and tightness conditions for the metric derivative, our experiments evaluate output dimensions only up to $d = 50$. How the tNTK behaves at substantially higher dimensions remains an open question.

Acknowledgments and Disclosure of Funding

We thank Jiequn Han for early discussions about distribution-aware active learning. This work was supported through the NYU IT High Performance Computing (HPC) resources, services, and staff expertise. Jayoung Ryu was partially supported by a Betty Moore Postdoctoral Fellowship, a Google TPU Research & Education Award. Romain Lopez and Jayoung Ryu were partially supported by a Schmidt Sciences AI2050 Fellowship Award. This work was supported by the Institute of Information Communications Technology Planning Evaluation (IITP) with a grant funded by the Ministry of Science and ICT (MSIT) of the Republic of Korea in connection with the Global AI Frontier Lab International Collaborative Research Agreement. (No. RS-2024-00469482 RS-2024-00509279) This work was partly supported in part by the NYUAD Center for Interdisciplinary Data Science AI (CIDSAI), funded by Tamkeen under the NYUAD Research Institute Award CG016.

Disclosures: Romain Lopez has been an employee of Genentech since August 31st, 2021.

References

[1] Ross D King, Kenneth E Whelan, Ffion M Jones, Philip GK Reiser, Christopher H Bryant, Stephen H Muggleton, Douglas B Kell, and Stephen G Oliver. Functional genomic hypothesis generation and experimentation by a robot scientist. *Nature*, 427(6971):247–252, 2004.

[2] Ross D King, Jem Rowland, Stephen G Oliver, Michael Young, Wayne Aubrey, Emma Byrne, Maria Liakata, Magdalena Markham, Pinar Pir, Larisa N Soldatova, et al. The automation of science. *Science*, 324(5923):85–89, 2009.

[3] Nathan J Szymanski, Bernardus Rendy, Yuxing Fei, Rishi E Kumar, Tanjin He, David Milsted, Matthew J McDermott, Max Gallant, Ekin Dogus Cubuk, Amil Merchant, Haegyeom Kim, Anubhav Jain, Christopher J Bartel, Kristin Persson, Yan Zeng, and Gerbrand Ceder. An autonomous laboratory for the accelerated synthesis of novel materials. *Nature*, 624(7990): 86–91, 2023.

[4] Christopher Tosh, Mauricio Tec, Jessica B White, Jeffrey F Quinn, Glorymar Ibanez Sanchez, Paul Calder, Andrew L Kung, Filemon S Dela Cruz, and Wesley Tansey. A Bayesian active learning platform for scalable combination drug screens. *Nature Communications*, 16(1):156, 2025.

[5] Nathan C Frey, Isidro Hötzel, Samuel D Stanton, Ryan Kelly, Robert G Alberstein, Emily K Makowski, Karolis Martinkus, Daniel Berenberg, Jack Bevers III, Tyler Bryson, et al. Lab-in-the-loop therapeutic antibody design with deep learning. *bioRxiv*, 2025.

[6] Jason Yang, Ravi G Lal, James C Bowden, Raul Astudillo, Mikhail A Hameedi, Sukhvinder Kaur, Matthew Hill, Yisong Yue, and Frances H Arnold. Active learning-assisted directed evolution. *Nature Communications*, 16(1):714, 2025.

[7] Kaiyi Jiang, Zhaoqing Yan, Matteo Di Bernardo, Samantha R Sgrizzi, Lukas Villiger, Alisan Kayabolen, BJ Kim, Josephine K Carscadden, Masahiro Hiraizumi, Hiroshi Nishimasu, et al. Rapid in silico directed evolution by a protein language model with EVOLVEpro. *Science*, 387 (6732):6006, 2024.

[8] Jennifer E Rood, Anna Hupalowska, and Aviv Regev. Toward a foundation model of causal cell and tissue biology with a perturbation cell and tissue atlas. *Cell*, 187(17):4520–4545, 2024.

[9] Jiaqi Zhang, Louis Cammarata, Chandler Squires, Themistoklis P Sapsis, and Caroline Uhler. Active learning for optimal intervention design in causal models. *Nature Machine Intelligence*, 5(10):1066–1075, 2023.

[10] Kexin Huang, Romain Lopez, Jan-Christian Hütter, Takamasa Kudo, Antonio Rios, and Aviv Regev. Sequential optimal experimental design of perturbation screens guided by multi-modal priors. In *Research in Computational Molecular Biology (RECOMB)*, pages 17–37. 2024.

[11] Jason Qin, Hans-Hermann Wessels, Carlos Fernandez-Granda, and Yuhan Hao. Active learning for efficient discovery of optimal combinatorial perturbations. In *International Conference on Machine Learning*, volume 267, pages 50363–50383, 13–19 Jul 2025.

[12] Benjamin DeMeo, Charlotte Nesbitt, Samuel A Miller, Daniel B Burkhardt, Inna Lipchina, Doris Fu, Peter Holderrieth, David Kim, Sergey Kolchenko, Artur Szalata, et al. Active learning framework leveraging transcriptomics identifies modulators of disease phenotypes. *Science*, 390(6776):8577, 2025.

[13] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018.

[14] Viraj Mehta, Syrine Belakaria, Vikramjeet Das, Ojash Neopane, Yijia Dai, Ilija Bogunovic, Barbara E Engelhardt, Stefano Ermon, Jeff Schneider, and Willie Neiswanger. Sample efficient preference alignment in LLMs via active exploration. In *Second Conference on Language Modeling*, 2025.

[15] William Muldrew, Peter Hayes, Mingtian Zhang, and David Barber. Active preference learning for large language models. In *International Conference on Machine Learning*, pages 36577–36590. PMLR, 2024.

[16] David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Machine learning*, 15(2):201–221, 1994.

[17] David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active learning with statistical models. *Journal of artificial intelligence research*, 4:129–145, 1996.

[18] Burr Settles. Active learning literature survey. Technical report, 2009.

[19] Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *International Conference on Learning Representations*, 2020.

[20] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv*, 2011.

[21] Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. BatchBALD: Efficient and diverse batch acquisition for deep Bayesian active learning. In *Advances in Neural Information Processing Systems*, 2019.

[22] Robert Pinsler, Jonathan Gordon, Eric T. Nalisnick, and José Miguel Hernández-Lobato. Bayesian batch active learning as sparse subset approximation. In *Advances in Neural Information Processing Systems*, 2019.

[23] Masashi Sugiyama. Active learning in approximately linear regression based on conditional expectation of generalization error. *Journal of Machine Learning Research*, 7(6):141–166, 2006.

[24] Roi Reichart, Katrin Tomanek, Udo Hahn, and Ari Rappoport. Multi-task active learning for linguistic annotations. In *Annual Meeting of the Association for Computational Linguistics*, pages 861–869, 2008.

[25] David Holzmüller, Viktor Zaverkin, Johannes Kästner, and Ingo Steinwart. A framework and benchmark for deep batch active learning for regression. *Journal of Machine Learning Research*, 24(164):1–81, 2023.

[26] Jordan T. Ash, Surbhi Goel, Akshay Krishnamurthy, and Sham M. Kakade. Gone fishing: Neural active learning with Fisher embeddings. In *Advances in Neural Information Processing Systems*, 2021.

[27] Wouter Meuleman, Alexander Muratov, Eric Rynes, Jessica Halow, Kristen Lee, Daniel Bates, Morgan Diegel, Douglas Dunn, Fidencio Neri, Athanasios Teodosiadis, Alex Reynolds, Eric Haugen, Jemma Nelson, Audra Johnson, Mark Frerker, Michael Buckley, Richard Sandstrom, Jeff Vierstra, Rajinder Kaul, and John Stamatoyannopoulos. Index and biological spectrum of human DNase I hypersensitive sites. *Nature*, 584(7820):244–251, 2020.

[28] Julien Bect, David Ginsbourger, Ling Li, Victor Picheny, and Emmanuel Vazquez. Sequential design of computer experiments for the estimation of a probability of failure. *Statistics and Computing*, 22(3):773–793, 2012.

[29] Themistoklis P Sapsis. Output-weighted optimal sampling for Bayesian regression and rare event statistics using few samples. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 476(2234), 2020.

[30] Charlotte Bunne, Andreas Krause, and Marco Cuturi. Supervised training of conditional Monge maps. In *Advances in Neural Information Processing Systems*, volume 35, pages 6859–6872, 2022.

[31] Alice Driessen, Benedek Harsanyi, Marianna Rapsomaniki, and Jannis Born. Towards generalizable single-cell perturbation modeling via the conditional Monge gap. *Nature Machine Intelligence*, 2026.

[32] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

[33] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

[34] Zheyuan Zhan, Defang Chen, Jian-Ping Mei, Zhenghe Zhao, Jiawei Chen, Chun Chen, Siwei Lyu, and Can Wang. Conditional image synthesis with diffusion models: A survey. *Transactions on Machine Learning Research*, September 2025.

[35] Xun Liang, Hanyu Wang, Yezhaohui Wang, Shichao Song, Jiawei Yang, Simin Niu, Jie Hu, Dan Liu, Shunyu Yao, Feiyu Xiong, and Zhiyu Li. Controllable text generation for large language models: A survey. *arXiv*, 2024.

[36] Charlotte Bunne, Stefan G Stark, Gabriele Gut, Jacobo Sarabia Del Castillo, Mitch Levesque, Kjong-Van Lehmann, Lucas Pelkmans, Andreas Krause, and Gunnar Rätsch. Learning single-cell perturbation responses using neural optimal transport. *Nature Methods*, 20(11):1759–1768, November 2023.

[37] Dominik Klein, Jonas Simon Fleck, Daniil Bobrovskiy, Lea Zimmermann, Sören Becker, Alessandro Palma, Leander Dony, Alejandro Tejada-Lapuerta, Guillaume Huguet, Hsiu-Chuan Lin, Nadezhda Azbukina, Fátima Sanchís-Calleja, Theo Uscidda, Artur Szalata, Manuel Gander, Aviv Regev, Barbara Treutlein, J Gray Camp, and Fabian J Theis. CellFlow enables generative single-cell phenotype modeling with flow matching. *bioRxiv*, 2025.

[38] Jayoung Ryu, Charlotte Bunne, Luca Pinello, Aviv Regev, and Romain Lopez. Cross-modality matching and prediction of perturbation responses with labeled Gromov-Wasserstein optimal transport. In *International Conference on Artificial Intelligence and Statistics*, pages 4258–4266, 2025.

[39] Martin Rohbeck, Edward De Brouwer, Charlotte Bunne, Jan-Christian Huetter, Anne Biton, Kelvin Y Chen, Aviv Regev, and Romain Lopez. Modeling complex system dynamics with flow matching across time and conditions. In *The Thirteenth International Conference on Learning Representations*, 2025.

[40] Lucas Ferreira DaSilva, Simon Senan, Judith F Kribelbauer-Swietek, Zain Munir Patel, Lithin Karmel Louis, Aniketh Janardhan Reddy, Sameer Gabbita, Jonathan D Rosen, Zach Nussbaum, César Miguel Valdez Córdova, Aaron Wenteler, Noah Weber, Tin M Tunjic, Martino Mansoldo, Talha Ahmad Khan, Gue-Ho Hwang, Vincent Gardeux, David T Humphreys, Cameron Smith, Matei Bejan, Peter Bromley, Will Connell, Bart Deplancke, Michael I Love, Emily S Wong, Wouter Meuleman, and Luca Pinello. Designing synthetic regulatory elements using the generative AI framework DNA-diffusion. *Nature Genetics*, 58(1):180–194, 2026.

[41] Brandon Amos, Lei Xu, and J. Zico Kolter. Input convex neural networks. In *International Conference on Machine Learning*, volume 70, pages 146–155, 2017.

[42] Théo Uscidda and Marco Cuturi. The Monge gap: a regularizer to learn all transport maps. In *International Conference on Machine Learning*, 2023.

[43] Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024.

[44] Vineeth Rakesh and Swayambhoo Jain. Efficacy of Bayesian neural networks in active learning. In *IEEE/CVF conference on computer vision and pattern recognition*, pages 2601–2609, 2021.

[45] Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: representing model uncertainty in deep learning. In *International Conference on Machine Learning*, volume 48, pages 1050–1059, 2016.

[46] Mohammad Emtiyaz Khan, Alexander Immer, Ehsan Abedi, and Maciej Korzepa. Approximate inference turns deep networks into Gaussian processes. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[47] Roman Novak, Jascha Sohl-Dickstein, and Samuel S Schoenholz. Fast finite width neural tangent kernel. In *International Conference on Machine Learning*, pages 17018–17044, 2022.

[48] Maurice Fréchet. Les éléments aléatoires de nature quelconque dans un espace distancié. *Annales de l'institut Henri Poincaré*, 10(4):215–310, 1948.

[49] Martial Agueh and Guillaume Carlier. Barycenters in the Wasserstein space. *SIAM Journal on Mathematical Analysis*, 43(2):904–924, 2011.

[50] David J. C. MacKay. A practical Bayesian framework for backpropagation networks. *Neural Computation*, 4(3):448–472, 1992.

[51] Erik Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, Matthias Bauer, and Philipp Hennig. Laplace redux – effortless Bayesian deep learning. In *Advances in Neural Information Processing Systems*, 2021.

[52] Wuchen Li and Jiaxi Zhao. Wasserstein information matrix. *Information Geometry*, 6(1): 203–255, 2023.

[53] Shu Liu, Wuchen Li, Hongyuan Zha, and Haomin Zhou. Neural parametric Fokker–Planck equation. *SIAM Journal on Numerical Analysis*, 60(3):1385–1449, 2022.

[54] Sang Jun Moon, Jong-June Jeon, Jason Sang Hun Lee, and Yongdai Kim. Learning multiple quantiles with neural networks. *Journal of Computational and Graphical Statistics*, 30(4): 1238–1248, 2021.

[55] NOAA National Centers for Environmental Information. Global surface summary of the day - gsod. https://www.ncei.noaa.gov/access/metadata/landing-page/bin/iso?id=gov.noaa.ncdc:C00516, 1999. Temperature data between 2023-2025.

[56] Thorben Werner and Lars Schmidt-Thieme. Bayesian active learning by distribution disagreement. *arXiv*, 2025.

[57] Ch Tsitouras. Runge–Kutta pairs of order 5 (4) satisfying only the first column simplifying assumption. *Computers & Mathematics with Applications*, 62(2):770–775, 2011.

[58] B. van der Pol. On "relaxation-oscillations". *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 7(2):978–992, 1926.

[59] Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. An evaluation dataset for intent classification and out-of-scope prediction. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1311–1316, 2019.

[60] CLINC150. UCI Machine Learning Repository, 2020. DOI: https://doi.org/10.24432/C5MP58.

[61] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. MINILM: deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in Neural Information Processing Systems*, 2020.

[62] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, 2020.

[63] Geoffrey Schiebinger, Jian Shu, Marcin Tabaka, Brian Cleary, Vidya Subramanian, Aryeh Solomon, Joshua Gould, Siyan Liu, Stacie Lin, Peter Berube, Lia Lee, Jenny Chen, Justin Brumbaugh, Philippe Rigollet, Konrad Hochedlinger, Rudolf Jaenisch, Aviv Regev, and Eric S Lander. Optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming. *Cell*, 176(4):928–943.e22, 2019.

[64] Sanjay R Srivatsan, José L McFaline-Figueroa, Vijay Ramani, Lauren Saunders, Junyue Cao, Jonathan Packer, Hannah A Pliner, Dana L Jackson, Riza M Daza, Lena Christiansen, Fan Zhang, Frank Steemers, Jay Shendure, and Cole Trapnell. Massively multiplex chemical transcriptomics at single-cell resolution. *Science*, 367(6473):45–51, 2020.

[65] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

[66] Christina Winkler, Daniel Worrall, Emiel Hoogeboom, and Max Welling. Learning likelihoods with conditional normalizing flows. *arXiv*, 2019.

[67] Kamalika Chaudhuri, Sham Kakade, Praneeth Netrapalli, and Sujay Sanghavi. Convergence rates of active learning for maximum likelihood estimation. *Advances in Neural Information Processing Systems*, 2012.

[68] Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient flows: In metric spaces and in the space of probability measures*. Lectures in Mathematics. Birkhauser Verlag AG, 2004.

[69] Cédric Villani et al. *Optimal transport: Old and New*, volume 338. Springer, 2009.

[70] Murray Rosenblatt. Remarks on a multivariate transformation. *The annals of mathematical statistics*, 23(3):470–472, 1952.

[71] Herbert Knothe. Contributions to the theory of convex bodies. *Michigan Mathematical Journal*, 4(1):39–52, 1957.

[72] Clark R. Givens and Rae Michael Shortt. A class of Wasserstein metrics for probability distributions. *Michigan Mathematical Journal*, 31(2):231–240, 1984.

[73] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11, 2018.

[74] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (GELUs). *arXiv*, 2016.

[75] Zoe Piran. sciplex3. 2023. doi: 10.6084/m9.figshare.22122701.v1.

[76] Romain Lopez, Jeffrey Regier, Michael B Cole, Michael I Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature Methods*, 15(12):1053–1058, December 2018.

[77] Takamasa Kudo, Romain Lopez, Ana M. Meireles, Antonio Rios, Paula Coelho, Vineethkrishna Chandrasekar, Dylan C. Lam, Jan-Christian Huetter, Mineto Ota, Orit Rozenblatt-Rosen, Levi Garraway, Kathryn Geiger-Shuller, Avtar Singh, Jonathan K. Pritchard, and Aviv Regev. Scalable multimodal mapping of macrophage regulatory architecture by integrating optical and transcriptomic pooled screens. *bioRxiv*, 2026.

[78] Ethan Weinberger, Chris Lin, and Su-In Lee. Isolating salient variations of interest in single-cell data with contrastiveVI. *Nature Methods*, 20(9):1336–1345, 2023.

[79] David Feldman, Avtar Singh, Jonathan L Schmid-Burgk, Rebecca J Carlson, Anja Mezger, Anthony J Garrity, Feng Zhang, and Paul C Blainey. Optical pooled screens in human cells. *Cell*, 179(3):787–799.e17, October 2019.

[80] Pedro C Álvarez Esteban, E del Barrio, J A Cuesta-Albertos, and C Matrán. A fixed-point approach to barycenters in Wasserstein space. *arXiv*, November 2015.

[81] Carsten Lüth, Till Bungert, Lukas Klein, and Paul Jaeger. Navigating the pitfalls of active learning evaluation: A systematic framework for meaningful performance assessment. In *Advances in Neural Information Processing Systems*, volume 36, pages 9789–9836, 2023.

Code Availability Statement

The code to reproduce the figures and tables, as well as to run the model can be found at https://github.com/biologicalML/transport_ntk.

Supplementary Materials

This appendix is organized as follows.

- Appendix A: Theoretical derivations for the motivation for distribution-aware active learning and the properties of the tNTK.
  - Appendix A.1: Proof of Proposition 1 on risk decomposition.
  - Appendix A.2: Proof of Lemma 1 on mean-variance decomposition of risk in one dimension.
  - Appendix A.3: Proof of Theorem 1 on the upper bound of the Fréchet variance.
  - Appendix A.4: Positive semi-definiteness of the tNTK.
  - Appendix A.5: Simplified kernel computation for models with a linear output layer, using sensitivity to last-layer parameters.
  - Appendix A.6: Proof of Lemma 2 on kernel decomposition for affine transport.
- Appendix B: Geometric interpretation of the tNTK as an upper bound on the metric derivative on the 2-Wasserstein space, and relationship to metric derivative on the Knothe–Rosenblatt distance.
  - Appendix B.1: Background on curves and metric derivatives on Wasserstein space.
  - Appendix B.2: The norm induced by $G_\theta(x)$ upper-bounds the metric derivative of a curve induced by parameter perturbation.
  - Appendix B.3: Analysis of the gap between the bound and the metric derivative, and conditions under which it is tight.
- Appendix C: Exact metric tensor under the Knothe–Rosenblatt distance: the tNTK recovers the squared metric derivative for lower-triangular affine maps.
- Appendix D: Experimental details.
  - Appendix D.1: Conditional histogram example from Section 3.2.
  - Appendix D.2: NTK baseline calculation.
  - Appendix D.3: Affine transport of 2D Gaussians (Section 5.1).
  - Appendix D.4: Conditional quantile prediction (Section 5.2).
  - Appendix D.5: Conditional flow matching (Section 5.3).
  - Appendix D.6: Conditional Monge gap (Section 5.4).
- Appendix E: Additional results.
  - Appendix E.1: Empirical calculation of gap of the upper bound in Theorem 1.
  - Appendix E.2: Active learning results on the higher-dimensional affine Gaussian transport.
  - Appendix E.3: Results on tNTK computation.
  - Appendix E.4: Additional baselines for the experiments presented in the main text.
- Appendix F: Supplementary Figures.

A Theory of the Transport Neural Tangent Kernel

A.1 Proof of Proposition 1

Proposition 1 (Mean-shape decomposition of risk)**.** *The Wasserstein risk defined in Equation (5) decomposes as*

$$\mathcal{R}(x,\mathcal{D}) = \underbrace{\mathbb{E}_{\theta|\mathcal{D}}\big[\|m_\theta(x)-m(x)\|^2\big]}_{\mathcal{R}_{mean}(x,\mathcal{D})} + \underbrace{\mathbb{E}_{\theta|\mathcal{D}}\big[W_2^2(\mu_\theta^c(x),\mu^c(x))\big]}_{\mathcal{R}_{shape}(x,\mathcal{D})}, \tag{6}$$

where $m_\rho = \int y\, d\rho(y)$ denotes the mean of ρ, and ρ^c is obtained by centering ρ for any $\rho \in \mathcal{P}_2(\mathcal{Y})$.

Proof. We drop the dependence on x for readability. For any vector $v \in \mathcal{Y}$, we denote by $\tau_v : \mathcal{Y} \to \mathcal{Y}$ the translation operator $\tau_v(y) = y - v$. For any measure $\rho \in \mathcal{P}_2(\mathcal{Y})$ with mean m_ρ, the centered measure is then $\rho^c = (\tau_{m_\rho})\#\rho$. We extend this notation to product spaces by writing $\tau_{v,v'}(y,y') = (y-v, y'-v')$ for the translation acting on couplings.

Let $\pi \in \Pi(\mu,\mu_\theta)$ be an optimal coupling for $W_2^2(\mu,\mu_\theta)$, and let $\pi^c \in \Pi(\mu^c,\mu_\theta^c)$ be an optimal coupling for $W_2^2(\mu^c,\mu_\theta^c)$. We will use the following algebraic identity twice:

$$\|(y-m)-(y'-m_\theta)\|^2 = \|y-y'\|^2 + \|m_\theta - m\|^2 + 2(m_\theta-m)^\top(y-y'). \tag{22}$$

Step 1: $(\tau_{m,m_\theta})_\#\pi$ is admissible for μ^c,μ_θ^c. Define $\hat\pi = (\tau_{m,m_\theta})_\#\pi$. Since π has marginals μ and μ_θ, the pushforward $\hat\pi$ has marginals $(\tau_m)\#\mu = \mu^c$ and $(\tau_{m_\theta})_\#\mu_\theta = \mu_\theta^c$, so $\hat\pi \in \Pi(\mu^c,\mu_\theta^c)$.

Applying (22) with $u = y-m$ and $u' = y'-m_\theta$, and integrating against π, using $\mathbb{E}_\pi[y] = m$ and $\mathbb{E}_\pi[y'] = m_\theta$:

$$\iint \|u-u'\|^2\, d\hat\pi(u,u') = \iint \|(y-m)-(y'-m_\theta)\|^2\, d\pi(y,y') \tag{23}$$

$$= \underbrace{\iint \|y-y'\|^2\, d\pi}_{=W_2^2(\mu,\mu_\theta)} + \|m_\theta-m\|^2 + 2(m_\theta-m)^\top \underbrace{\mathbb{E}_\pi[y-y']}_{=m-m_\theta} \tag{24}$$

$$= W_2^2(\mu,\mu_\theta) - \|m_\theta-m\|^2. \tag{25}$$

Since $\hat\pi$ is admissible for μ^c,μ_θ^c but not necessarily optimal,

$$W_2^2(\mu^c,\mu_\theta^c) \le W_2^2(\mu,\mu_\theta) - \|m_\theta-m\|^2. \tag{26}$$

Step 2: $(\tau_{-m,-m_\theta})_\#\pi^c$ is admissible for μ,μ_θ. Symmetrically, define $\tilde\pi = (\tau_{-m,-m_\theta})\#\pi^c$. Since π^c has marginals μ^c and μ_θ^c, the pushforward $\tilde\pi$ has marginals μ and μ_θ, so $\tilde\pi \in \Pi(\mu,\mu_\theta)$.

Applying (22) with $y = u+m$ and $y' = u'+m_\theta$, and integrating against π^c, using $\mathbb{E}_{\pi^c}[u] = 0$ and $\mathbb{E}_{\pi^c}[u'] = 0$:

$$\iint \|y-y'\|^2\, d\tilde\pi(y,y') = \iint \|(u+m)-(u'+m_\theta)\|^2\, d\pi^c(u,u') \tag{27}$$

$$= \underbrace{\iint \|u-u'\|^2\, d\pi^c}_{=W_2^2(\mu^c,\mu_\theta^c)} + \|m_\theta-m\|^2 + 2(m_\theta-m)^\top \underbrace{\mathbb{E}_{\pi^c}[u'-u]}_{=0} \tag{28}$$

$$= W_2^2(\mu^c,\mu_\theta^c) + \|m_\theta-m\|^2. \tag{29}$$

Since $\tilde\pi$ is admissible for μ,μ_θ but not necessarily optimal,

$$W_2^2(\mu,\mu_\theta) \le W_2^2(\mu^c,\mu_\theta^c) + \|m_\theta-m\|^2. \tag{30}$$

Step 3: Conclusion. Combining (26) and (30),

$$W_2^2(\mu,\mu_\theta) = W_2^2(\mu^c,\mu_\theta^c) + \|m_\theta-m\|^2. \tag{31}$$

Reinstating the dependence on x and taking expectations over $\theta \mid \mathcal{D}$:

$$\mathcal{R}(x,\mathcal{D}) = \mathbb{E}_{\theta|\mathcal{D}}\big[W_2^2(\mu_\theta(x),\mu(x))\big] \tag{32}$$

$$= \mathbb{E}_{\theta|\mathcal{D}}\big[\|m_\theta(x)-m(x)\|^2\big] + \mathbb{E}_{\theta|\mathcal{D}}\big[W_2^2(\mu_\theta^c(x),\mu^c(x))\big] \tag{33}$$

$$= \mathcal{R}_{\text{mean}}(x,\mathcal{D}) + \mathcal{R}_{\text{shape}}(x,\mathcal{D}), \tag{34}$$

which proves the proposition. □

A.2 Proof of Lemma 1

Lemma 1 (1D posterior risk)**.** *Assume the model is unbiased at the mean and every quantile level, i.e.,* $\mathbb{E}_{\theta|\mathcal{D}}[m_\theta(x)] = m(x)$ *and* $\mathbb{E}_{\theta|\mathcal{D}}[F_\theta^{-1}(x;\tau)] = F^{-1}(x;\tau)$ *for all* x, τ*. Then*

$$\mathcal{R}_{mean}(x,\mathcal{D}) = \mathrm{Var}_{\theta|\mathcal{D}}[m_\theta(x)], \qquad \mathcal{R}_{shape}(x,\mathcal{D}) = \int_0^1 \mathrm{Var}_{\theta|\mathcal{D}}[g_\theta(x;\tau)]\,d\tau, \tag{7}$$

where $g_\theta(x;\tau) = F_\theta^{-1}(x;\tau) - m_\theta(x)$ *denotes the centered quantile function of* $\mu_\theta(x)$*.*

Proof. We drop the dependence on x for readability. Throughout, all variances, covariances, and expectations are taken with respect to the posterior $\theta \mid \mathcal{D}$.

Step 1: Quantile representation of W_2^2. For one-dimensional distributions, the squared 2-Wasserstein distance admits the closed-form quantile representation:

$$W_2^2(\mu_\theta,\mu) = \int_0^1 \left(F_\theta^{-1}(\tau) - F^{-1}(\tau)\right)^2 d\tau, \tag{35}$$

where we remind the reader that $F^{-1}(\tau)$ (resp. $F_\theta^{-1}(\tau)$) corresponds to the τ quantile of the distribution μ (resp. μ_θ).

Step 2: Derivation of the mean risk. Under the assumption $\mathbb{E}[m_\theta] = m$, the squared error reduces to the posterior variance:

$$\mathcal{R}_{\text{mean}}(x,\mathcal{D}) = \mathbb{E}\left[(m_\theta - m)^2\right] = \mathrm{Var}[m_\theta]. \tag{36}$$

Step 3: Derivation of the shape risk. The centered measures μ_θ^c and μ^c have quantile functions $g_\theta(\tau) = F_\theta^{-1}(\tau) - m_\theta$ and $g(\tau) = F^{-1}(\tau) - m$, respectively. Applying (35) to the centered measures:

$$W_2^2(\mu_\theta^c,\mu^c) = \int_0^1 (g_\theta(\tau) - g(\tau))^2\,d\tau. \tag{37}$$

Taking expectations and exchanging the integral and expectation by Fubini's theorem:

$$\mathcal{R}_{\text{shape}}(x,\mathcal{D}) = \int_0^1 \mathbb{E}\left[(g_\theta(\tau) - g(\tau))^2\right] d\tau. \tag{38}$$

Because we have that $\mathbb{E}[m_\theta] = m$ and that for all $\tau \in [0,1]$, $\mathbb{E}[F_\theta^{-1}(\tau)] = F^{-1}(\tau)$, it implies that $\mathbb{E}[g_\theta(\tau)] = g(\tau)$ by linearity of the expectation. Therefore, the squared error reduces to the posterior variance:

$$\mathbb{E}\left[(g_\theta(\tau) - g(\tau))^2\right] = \mathrm{Var}[g_\theta(\tau)], \tag{39}$$

and consequently

$$\mathcal{R}_{\text{shape}}(x,\mathcal{D}) = \int_0^1 \mathrm{Var}[g_\theta(\tau)]\,d\tau. \tag{40}$$

Combining (36) and (40) proves the lemma. □

A.3 Proof of Theorem 1

Theorem 1 (Upper bound on the posterior Fréchet variance)**.** *Let* $\theta \mid \mathcal{D} \sim \mathcal{N}(\hat{\theta}, \Sigma)$*, with* $\Sigma \in \mathbb{R}^{p\times p}$ *positive definite and* $\|\Sigma\|_{\text{op}} < 1$*. Assume that for every* $x \in \mathcal{X}$ *and for* ν*-almost every* $y \in \mathcal{Y}$*, the map* $\theta \mapsto T_\theta(x)(y)$ *is* $C^2(\mathbb{R}^p)$*, and that the Jacobian* $D_\theta T_{\hat{\theta}}(x)(\cdot)$ *and the pointwise supremum of the operator norm of the Hessian,* $\sup_\theta \|D_\theta^2 T_\theta(x)(\cdot)\|_{\text{op}}$*, are in* $L^2(\nu)$*. Then, there exists* $C(x)$ *such that:*

$$V_F(x) \le \mathrm{Tr}\left(G_{\hat{\theta}}(x)\,\Sigma\right) + C(x)\|\Sigma\|_{\text{op}}^{3/2}, \tag{11}$$

where the $p \times p$ *parameter sensitivity matrix at input* x *is*

$$G_{\hat{\theta}}(x) = \int D_\theta T_{\hat{\theta}}(x)(y)^\top D_\theta T_{\hat{\theta}}(x)(y)\,d\nu(y). \tag{12}$$

Inequality (11) *is tight to leading order when* $\dim(\mathcal{Y}) = 1$ *and* $T_\theta(x)$ *is monotone in* y*.*

Proof. The proof has three parts: (i) we reduce $V_F(x)$ to an integrated $L^2(\nu)$ distance between transport maps, (ii) we expand the integrand in Σ and bound each term and collect the bounds, (iii) we analyze tightness.

Part 1: Reduction to an $L^2(\nu)$ functional

Reducing V_F to a single-draw expectation. By definition of the infimum, taking $\rho = \mu_{\hat{\theta}}(x)$ in (10) yields

$$V_F(x) \le \mathbb{E}_{\theta|\mathcal{D}}\big[W_2^2(\mu_\theta(x), \mu_{\hat{\theta}}(x))\big]. \tag{41}$$

The remainder of the proof bounds the right-hand side.

Synchronized coupling. We momentarily drop the dependence on x and derive an upper bound on $W_2^2(\mu_\theta, \mu_{\hat{\theta}})$ for fixed θ. Let $\Phi_{\theta,\hat{\theta}} : \mathcal{Y} \to \mathcal{Y} \times \mathcal{Y}$ denote the map $y \mapsto (T_\theta(y), T_{\hat{\theta}}(y))$, and introduce the synchronized coupling $\pi = (\Phi_{\theta,\hat{\theta}})_{\#}\nu$. The measure π is a coupling of μ_θ and $\mu_{\hat{\theta}}$: for any test function φ, the first marginal satisfies

$$\int \varphi(u)\, d\pi_1(u) = \int \varphi(T_\theta(y))\, d\nu(y) = \int \varphi(u)\, d\mu_\theta(u), \tag{42}$$

where we applied the pushforward identity twice, using $\mu_\theta = T_\theta \# \nu$. The same argument yields $\pi_2 = \mu_{\hat{\theta}}$, so $\pi \in \Pi(\mu_\theta, \mu_{\hat{\theta}})$. Applying the pushforward identity to the cost function $(u, u') \mapsto \|u - u'\|^2$ gives the transport cost of π:

$$\int \|u - u'\|^2\, d\pi(u, u') = \int \|T_\theta(y) - T_{\hat{\theta}}(y)\|^2\, d\nu(y). \tag{43}$$

Because W_2^2 is the infimum of the transport cost over all admissible couplings,

$$W_2^2(\mu_\theta, \mu_{\hat{\theta}}) \le \int \|T_\theta(y) - T_{\hat{\theta}}(y)\|^2\, d\nu(y). \tag{44}$$

Reinstating the dependence on x and combining with (41),

$$V_F(x) \le \mathbb{E}_{\theta|\mathcal{D}}\Big[\int \|T_\theta(x)(y) - T_{\hat{\theta}}(x)(y)\|^2\, d\nu(y)\Big]. \tag{45}$$

Taylor expansion around $\hat{\theta}$. We continue to drop the dependence on x and write $J(y) := D_\theta T_{\hat{\theta}}(y)$ for the Jacobian at the posterior mean. By the smoothness assumption, for ν-a.e. y and all $\theta \in \mathbb{R}^p$, a second-order Taylor expansion gives

$$T_\theta(y) = T_{\hat{\theta}}(y) + J(y)(\theta - \hat{\theta}) + r(\theta, y), \tag{46}$$

where the remainder satisfies

$$\|r(\theta, y)\| \le \tfrac{1}{2}\,\kappa(y)\,\|\theta - \hat{\theta}\|^2, \qquad \kappa(y) := \sup_\theta \|D_\theta^2 T_\theta(y)\|_{\mathrm{op}}, \tag{47}$$

and $\kappa \in L^2(\nu)$ by assumption. Squaring and expanding the inner product,

$$\|T_\theta(y) - T_{\hat{\theta}}(y)\|^2 = \underbrace{\|J(y)(\theta - \hat{\theta})\|^2}_{=:A(y,\theta)} + \underbrace{2\,\langle J(y)(\theta - \hat{\theta}),\, r(\theta, y)\rangle}_{=:B(y,\theta)} + \underbrace{\|r(\theta, y)\|^2}_{=:C(y,\theta)}. \tag{48}$$

Part 2: Term-by-term bounds

Leading term. Since $\theta \sim \mathcal{N}(\hat{\theta}, \Sigma)$, we have $\mathbb{E}[(\theta - \hat{\theta})(\theta - \hat{\theta})^\top] = \Sigma$. Using the cyclicity of the trace,

$$\mathbb{E}[A(y, \theta)] = \mathbb{E}\big[(\theta - \hat{\theta})^\top J(y)^\top J(y)\,(\theta - \hat{\theta})\big] = \mathrm{Tr}\big(J(y)^\top J(y)\,\Sigma\big). \tag{49}$$

By linearity of the trace operator,

$$\int \mathbb{E}[A(y, \theta)]\, d\nu(y) = \mathrm{Tr}\Big(\Big(\int J(y)^\top J(y)\, d\nu(y)\Big)\Sigma\Big) = \mathrm{Tr}\big(G_{\hat{\theta}}(x)\,\Sigma\big). \tag{50}$$

Cross term. By Cauchy–Schwarz applied pointwise in y and then in $L^2(\mathbb{P})$,

$$\mathbb{E}|B(y, \theta)| \le 2\,\mathbb{E}\big[\|J(y)(\theta - \hat{\theta})\|^2\big]^{1/2}\ \mathbb{E}\big[\|r(\theta, y)\|^2\big]^{1/2}. \tag{51}$$

The first factor equals $\sqrt{\mathrm{Tr}(J(y)^\top J(y)\,\Sigma)}$ by (49). For the second factor, applying (47),

$$\mathbb{E}\big[\|r(\theta, y)\|^2\big] \le \tfrac{1}{4}\,\kappa(y)^2\,\mathbb{E}\big[\|\theta - \hat{\theta}\|^4\big]. \tag{52}$$

For $\theta \sim \mathcal{N}(\hat{\theta}, \Sigma)$, the standard fourth-moment identity gives $\mathbb{E}\|\theta - \hat{\theta}\|^4 = (\operatorname{Tr}\Sigma)^2 + 2\operatorname{Tr}(\Sigma^2) \le 3(\operatorname{Tr}\Sigma)^2 \le 3p^2\|\Sigma\|_{\mathrm{op}}^2$, so

$$\mathbb{E}\big[\|r(\theta, y)\|^2\big] \le \tfrac{3}{4}\, p^2\, \kappa(y)^2\, \|\Sigma\|_{\mathrm{op}}^2. \tag{53}$$

Substituting into (51) and integrating over ν via Cauchy–Schwarz,

$$\int \mathbb{E}|B(y,\theta)|\, d\nu(y) \le \sqrt{3}\, p\, \|\Sigma\|_{\mathrm{op}} \int \kappa(y) \sqrt{\operatorname{Tr}(J(y)^\top J(y)\, \Sigma)}\, d\nu(y) \tag{54}$$

$$\le \sqrt{3}\, p\, \|\Sigma\|_{\mathrm{op}}\, \|\kappa\|_{L^2(\nu)} \sqrt{\operatorname{Tr}(G_{\hat{\theta}}(x)\, \Sigma)} \tag{55}$$

$$\le \sqrt{3}\, p\, \|\kappa\|_{L^2(\nu)} \sqrt{\operatorname{Tr}(G_{\hat{\theta}}(x))}\, \|\Sigma\|_{\mathrm{op}}^{3/2}, \tag{56}$$

where the last step uses $\operatorname{Tr}(G_{\hat{\theta}}(x)\,\Sigma) \le \operatorname{Tr}(G_{\hat{\theta}}(x))\, \|\Sigma\|_{\mathrm{op}}$ (since $G_{\hat{\theta}}(x)$ is positive semi-definite) and $\operatorname{Tr}(G_{\hat{\theta}}(x)) < \infty$ by the integrability assumption.

Quadratic remainder. Using (53) and integrating over ν,

$$\int \mathbb{E}[C(y,\theta)]\, d\nu(y) \le \tfrac{3}{4}\, p^2\, \|\kappa\|_{L^2(\nu)}^2\, \|\Sigma\|_{\mathrm{op}}^2 \le \tfrac{3}{4}\, p^2\, \|\kappa\|_{L^2(\nu)}^2\, \|\Sigma\|_{\mathrm{op}}^{3/2}, \tag{57}$$

where the last step uses $\|\Sigma\|_{\mathrm{op}} < 1$.

Combining the bounds. Combining (45), (48), (50), (56), and (57),

$$V_F(x) \le \int \mathbb{E}[A(y,\theta) + B(y,\theta) + C(y,\theta)]\, d\nu(y) \tag{58}$$

$$\le \operatorname{Tr}\left(G_{\hat{\theta}}(x)\, \Sigma\right) + \sqrt{3}\, p\, \|\kappa\|_{L^2(\nu)} \sqrt{\operatorname{Tr}(G_{\hat{\theta}}(x))}\, \|\Sigma\|_{\mathrm{op}}^{3/2} + \tfrac{3}{4}\, p^2\, \|\kappa\|_{L^2(\nu)}^2\, \|\Sigma\|_{\mathrm{op}}^{3/2} \tag{59}$$

$$= \operatorname{Tr}\left(G_{\hat{\theta}}(x)\, \Sigma\right) + C(x)\, \|\Sigma\|_{\mathrm{op}}^{3/2}, \tag{60}$$

where

$$C(x) = \sqrt{3}\, p\, \|\kappa\|_{L^2(\nu)} \sqrt{\operatorname{Tr}(G_{\hat{\theta}}(x))} + \tfrac{3}{4}\, p^2\, \|\kappa\|_{L^2(\nu)}^2, \tag{61}$$

which is the claimed bound.

Part 3: The 1D monotone case

We show that when $\dim(\mathcal{Y}) = 1$ and $T_\theta(x)$ is monotone in y, the inequality (11) becomes tight in the leading order. In this setting, $V_F(x)$ admits a closed-form expression that we compute directly.

We drop the dependence on x throughout, and write $Q_\theta(\tau) := F_{\mu_\theta}^{-1}(\tau)$ for the quantile function of μ_θ. Monotonicity of T_θ gives the quantile representation $T_\theta = Q_\theta \circ F_\nu$, and the change of variables $\tau = F_\nu(y)$ yields the isometry

$$W_2^2(\mu_\theta, \mu_{\theta'}) = \int_0^1 \big(Q_\theta(\tau) - Q_{\theta'}(\tau)\big)^2\, d\tau = \|Q_\theta - Q_{\theta'}\|_{L^2[0,1]}^2. \tag{62}$$

The Wasserstein-2 geometry on $\{\mu_\theta\}$ is therefore the Euclidean geometry of $L^2[0,1]$ on $\{Q_\theta\}$, and standard L^2 identities give a closed form for the Fréchet barycenter and variance. The barycenter quantile function is $\bar{Q}(\tau) := \mathbb{E}_{\theta|\mathcal{D}}[Q_\theta(\tau)]$, and the Fréchet variance is

$$V_F(x) = \mathbb{E}_{\theta|\mathcal{D}}\big[\|Q_\theta - \bar{Q}\|_{L^2[0,1]}^2\big] = \int_0^1 \operatorname{Var}_{\theta|\mathcal{D}}[Q_\theta(\tau)]\, d\tau, \tag{63}$$

by Fubini and the L^2 bias-variance decomposition.

The argument of Parts 1–2 now applies pointwise in τ to $Q_\theta(\tau)$ instead of to $T_\theta(y)$, with the synchronized-coupling step replaced by the isometry (62) (which is exact rather than an inequality). Because Equation 50 is an equality, the upper bound from Theorem 1 is tight at leading order. More precisely, when $\|\Sigma\|_{\mathrm{op}} \to 0$:

$$V_F(x) = \operatorname{Tr}\left(G_{\hat{\theta}}(x)\Sigma\right) + O(\|\Sigma\|_{\mathrm{op}}^{3/2}). \tag{64}$$

□

A.4 Positive semi-definiteness of the tNTK

Proposition 2. *The tNTK $k_\theta : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defined in Equation 13 is positive semi-definite.*

Proof. Let $n \in \mathbb{N}^*$, and $x_1, \dots, x_n \in \mathcal{X}$ denote n input points. Let $c_1, \dots, c_n \in \mathbb{R}$ be weights. Writing $J_x(y) = D_\theta T_\theta(x)(y) \in \mathbb{R}^{d \times p}$ for the parameter Jacobian, we have that:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} c_i c_j \, k_\theta(x_i, x_j) = \sum_{i,j} c_i c_j \int \mathrm{Tr}\big[J_{x_i}(y)^\top J_{x_j}(y)\big] \, d\nu(y) \tag{65}$$

$$= \int \sum_{i,j} c_i c_j \left\langle J_{x_i}(y),\, J_{x_j}(y) \right\rangle_F d\nu(y) \tag{66}$$

$$= \int \left\| \sum_{i=1}^{n} c_i \, J_{x_i}(y) \right\|_F^2 d\nu(y) \geq 0, \tag{67}$$

where $\langle \cdot, \cdot \rangle_F$ and $\| \cdot \|_F$ denote the Frobenius inner product and norm. The final inequality holds because the squared Frobenius norm is non-negative and ν is a positive measure. □

A.5 Closed-form expression for the last-layer tNTK

Proposition 3. *Let the transport map have a linear last layer, $T_\theta(x)(y) = W^\top h_\theta(x, y)$, where $h_\theta(x, y) \in \mathbb{R}^m$ is the penultimate-layer output and $W \in \mathbb{R}^{m \times d}$ is the last-layer weight matrix. Then the tNTK (13) restricted to the last-layer parameters admits the closed form:*

$$k_\theta^{\mathrm{ll}}(x, x') = d \int h_\theta(x, y)^\top h_\theta(x', y) \, d\nu(y). \tag{68}$$

Proof. For $(i, j, k) \in [m] \times [d] \times [d]$, the last-layer Jacobian satisfies

$$\frac{\partial T_\theta^k(x)(y)}{\partial W_{ij}} = \delta_{kj} \, h_\theta^i(x, y), \tag{69}$$

where δ_{kj} is the Kronecker delta and superscripts denote scalar components: $T_\theta^k(x)(y)$ is the k-th component of $T_\theta(x)(y)$, and $h_\theta^i(x, y)$ is the i-th component of $h_\theta(x, y)$. Substituting into (13):

$$k_\theta^{\mathrm{ll}}(x, x') = \int \sum_{i,j,k} \delta_{kj} h_\theta^i(x, y) \, \delta_{kj} h_\theta^i(x', y) \, d\nu(y) \tag{70}$$

$$= \int \sum_{i,j} h_\theta^i(x, y) \, h_\theta^i(x', y) \, d\nu(y) \tag{71}$$

$$= d \int h_\theta(x, y)^\top h_\theta(x', y) \, d\nu(y), \tag{72}$$

where the second equality uses $\sum_k \delta_{kj}^2 = 1$, and the third uses that the summand is independent of j, so the sum across j contributes a factor of d. □

A.6 Proof of Lemma 2

Lemma 2. *Let us assume that $T_\theta(x)$ is an affine map: $T_\theta(x)(y) = b_\theta(x) + M_\theta(x)y$ and that the base measure ν is centered and isotropic (i.e., $\mathbb{E}_{y\sim\nu}[y] = 0$ and $\mathbb{E}_{y\sim\nu}[yy^\top] = I$). Then*

$$k_\theta(x, x') = \mathrm{Tr}\left[D_\theta b_\theta(x)^\top D_\theta b_\theta(x')\right] + \sum_{j=1}^{d} \mathrm{Tr}\left[D_\theta M_\theta^j(x)^\top D_\theta M_\theta^j(x')\right], \tag{16}$$

where $M_\theta^j(x)$ denotes the j-th column of $M_\theta(x)$.

Proof. For an affine map $T_\theta(x)(y) = b_\theta(x) + M_\theta(x)y$, the parameter Jacobian is linear in y:

$$D_\theta T_\theta(x)(y) = D_\theta b_\theta(x) + \sum_{j=1}^{d} y_j \, D_\theta M_\theta^j(x), \tag{73}$$

where $M_\theta^j(x) \in \mathbb{R}^d$ is the j-th column of $M_\theta(x)$ and y_j is the j-th component of y. Substituting (73) into the kernel definition (13) and expanding the inner product, we get four groups of terms:

$$k_\theta(x,x') = \int \mathrm{Tr}[A(x,x') + B(x,x';y) + B(x',x;y) + C(x,x';y)]\, d\nu(y), \tag{74}$$

where

$$\begin{aligned} A(x,x') &= D_\theta b_\theta(x)^\top D_\theta b_\theta(x'), \\ B(x,x';y) &= D_\theta b_\theta(x)^\top \sum_{j=1}^d y_j\, D_\theta M_\theta^j(x'), \\ C(x,x';y) &= \left(\sum_{j=1}^d y_j\, D_\theta M_\theta^j(x)\right)^\top \left(\sum_{j'=1}^d y_{j'}\, D_\theta M_\theta^{j'}(x')\right). \end{aligned}$$

We compute each term in turn.

Constant term A. $A(x,x')$ is independent of y, so the integral leaves it unchanged:

$$\int \mathrm{Tr}[A(x,x')]\, d\nu(y) = \mathrm{Tr}\Big[D_\theta b_\theta(x)^\top D_\theta b_\theta(x')\Big]. \tag{75}$$

Cross terms B. The cross terms are linear in y, so under the centering assumption $\mathbb{E}[y] = 0$,

$$\int \mathrm{Tr}[B(x,x';y)]\, d\nu(y) = \mathrm{Tr}\left[D_\theta b_\theta(x)^\top \sum_{j=1}^d \underbrace{\mathbb{E}[y_j]}_{=0}\, D_\theta M_\theta^j(x')\right] = 0, \tag{76}$$

and similarly $\int \mathrm{Tr}[B(x',x;y)]\, d\nu(y) = 0$.

Quadratic term C. Expanding the product and using the trace's linearity,

$$\mathrm{Tr}[C(x,x';y)] = \sum_{j,j'=1}^d y_j y_{j'}\, \mathrm{Tr}\Big[D_\theta M_\theta^j(x)^\top D_\theta M_\theta^{j'}(x')\Big]. \tag{77}$$

Integrating against ν and using the isotropy assumption $\mathbb{E}[yy^\top] = I$, i.e., $\mathbb{E}[y_j y_{j'}] = \delta_{jj'}$:

$$\int \mathrm{Tr}[C(x,x';y)]\, d\nu(y) = \sum_{j,j'=1}^d \delta_{jj'}\, \mathrm{Tr}\Big[D_\theta M_\theta^j(x)^\top D_\theta M_\theta^{j'}(x')\Big] = \sum_{j=1}^d \mathrm{Tr}\Big[D_\theta M_\theta^j(x)^\top D_\theta M_\theta^j(x')\Big]. \tag{78}$$

Combining the terms. Summing the contributions of A, B, $B^\top$,

$$k_\theta(x,x') = \mathrm{Tr}[D_\theta b_\theta(x)^\top D_\theta b_\theta(x')] + \sum_{j=1}^d \mathrm{Tr}[D_\theta M_\theta^j(x)^\top D_\theta M_\theta^j(x')]. \tag{79}$$

□

B Geometry of parameter perturbations on the Wasserstein manifold

The parameter sensitivity matrix $G_{\hat{\theta}}(x)$ that appears in Theorem 1 admits a geometric interpretation as an (approximate) metric tensor on the Wasserstein manifold. This appendix makes this interpretation precise. We view a parameter perturbation $h \in \mathbb{R}^p$ as inducing a curve $t \mapsto \mu_{\theta+th}(x)$ on the 2-Wasserstein space $W_2(\mathcal{Y})$ and study its metric derivative at $t = 0$. Section B.1 reviews the necessary background on metric derivatives, absolutely continuous curves, and the tangent bundle of W_2. Section B.2 bounds the squared metric derivative of this curve by $h^\top G_\theta(x)h$, providing an alternative derivation of $G_\theta(x)$ via a different proof technique than Theorem 1. Section B.3 characterizes when this bound is tight: in dimension one and for monotone maps it is always exact, and in higher dimensions the gap is governed by the divergence-free component of the velocity field. The latter result points to a concrete avenue for constructing a tighter proxy in future work.

B.1 Background on metric derivatives in Wasserstein space

We restate the key facts about absolutely continuous curves in Wasserstein space and their characterization via the continuity equation. We work throughout in $W_2(\mathcal{Y})$ with $\mathcal{Y} = \mathbb{R}^d$. We need three preliminary objects: absolutely continuous curves, the metric derivative, and the tangent space of $\mathcal{P}_2(\mathcal{Y})$ at a measure μ.

Definition 1 (Absolutely continuous curve)**.** *A curve $(\mu_t)_{t\in(a,b)} : (a,b) \to \mathcal{P}_2(\mathcal{Y})$ is absolutely continuous of second order on the open interval (a,b) if there exists $m \in L^2(a,b)$ such that*

$$W_2(\mu_s,\mu_t) \le \int_s^t m(r)\,dr \qquad \forall\, a < s \le t < b. \tag{80}$$

The set of such curves is denoted by $AC^2(a,b;\mathcal{P}_2(\mathcal{Y}))$.

Definition 2 (Metric derivative)**.** *Let $(\mu_t)_{t\in(a,b)} \in AC^2(a,b;\mathcal{P}_2(\mathcal{Y}))$. The* metric derivative *at $t \in (a,b)$ is*

$$|\mu'|(t) := \lim_{s\to t} \frac{W_2(\mu_s,\mu_t)}{|s-t|}, \tag{81}$$

whenever this limit exists, and the upper metric derivative *is*

$$|\mu'|^+(t) := \limsup_{s\to t} \frac{W_2(\mu_s,\mu_t)}{|s-t|}, \tag{82}$$

which is always defined and coincides with $|\mu'|(t)$ whenever the latter exists.

Definition 3 (Tangent space)**.** *Let $\mu \in \mathcal{P}_2(\mathcal{Y})$. The* tangent space *of $\mathcal{P}_2(\mathcal{Y})$ at μ is*

$$\mathrm{Tan}_\mu \mathcal{P}_2(\mathcal{Y}) := \overline{\{\nabla\phi : \phi \in C_c^\infty(\mathcal{Y})\}}^{L^2(\mu;\mathbb{R}^d)}, \tag{83}$$

where $C_c^\infty(\mathcal{Y})$ denotes the space of smooth functions with compact support, and the closure is taken in $L^2(\mu;\mathbb{R}^d)$.

Results from Ambrosio et al. [68] tie these definitions together. First, the metric derivative of an absolutely continuous curve exists almost everywhere and is its minimal speed.

Proposition 4 (68, Theorem 1.1.2)**.** *Let $(\mu_t)_{t\in(a,b)} \in AC^2(a,b;\mathcal{P}_2(\mathcal{Y}))$. Then $|\mu'|(t)$ exists for a.e. $t \in (a,b)$, and belongs to $L^2(a,b)$. Moreover, the function $t \mapsto |\mu'|(t)$ satisfies (80), and $|\mu'|(t) \le m(t)$ for a.e. t for any other m satisfying (80). In this sense, $|\mu'|$ is the smallest admissible speed bound.*

Second, absolutely continuous curves are exactly those driven by a velocity field of square-integrable norm through the *continuity equation*. Every absolutely continuous curve admits such a field:

Theorem 2 (68, Theorem 8.3.1)**.** *Let $(\mu_t)_{t\in(a,b)} \in AC^2(a,b;\mathcal{P}_2(\mathcal{Y}))$. Then there exists a Borel vector field $v\colon (y,t) \mapsto v_t(y)$ such that*

$$v_t \in L^2(\mu_t;\mathbb{R}^d), \qquad \|v_t\|_{L^2(\mu_t)} \le |\mu'|(t) \quad \text{for a.e. } t \in (a,b), \tag{84}$$

and the continuity equation

$$\partial_t \mu_t + \nabla\cdot(v_t\mu_t) = 0 \qquad \text{in } \mathcal{Y}\times(a,b) \tag{85}$$

holds in the distributional sense. Moreover, $v_t \in \mathrm{Tan}_{\mu_t}\mathcal{P}_2(\mathcal{Y})$ for a.e. $t \in (a,b)$.

Conversely, any curve driven by a velocity field with square-integrable norm is absolutely continuous, and the norm of that field bounds the metric derivative:

Proposition 5 (68, Theorem 8.3.1)**.** *Let $(\mu_t)_{t\in(a,b)}$ be a curve in $\mathcal{P}_2(\mathcal{Y})$ that is continuous with respect to W_2 and satisfies (85) for some Borel vector field v_t with $\int_a^b \|v_t\|^2_{L^2(\mu_t)}\,dt < \infty$. Then $(\mu_t) \in AC^2(a,b;\mathcal{P}_2(\mathcal{Y}))$ and*

$$|\mu'|(t) \le \|v_t\|_{L^2(\mu_t)} \qquad \text{for a.e. } t \in (a,b). \tag{86}$$

Third, the velocity field driving a curve is not unique: adding any field w_t with $\nabla\cdot(w_t\mu_t) = 0$ leaves (85) unchanged. By Proposition 5, every such field satisfies $\|v_t\|_{L^2(\mu_t)} \ge |\mu'|(t)$ for a.e. t. The next result characterizes the fields achieving equality: they are exactly the tangent ones, and they are unique.

Proposition 6 (68, Proposition 8.4.5)**.** *Let $(\mu_t)_{t\in(a,b)} \in AC^2(a,b;\mathcal{P}_2(\mathcal{Y}))$ and let $v_t \in L^2(\mu_t;\mathbb{R}^d)$ be a vector field solving (85). Then*

$$\|v_t\|_{L^2(\mu_t)} = |\mu'|(t) \quad \text{for a.e. } t \in (a,b) \tag{87}$$

if and only if $v_t \in \mathrm{Tan}_{\mu_t}\mathcal{P}_2(\mathcal{Y})$ for a.e. $t \in (a,b)$. Moreover, such a v_t is uniquely determined for a.e. t.

We use this characterization in Lemma 4 to identify the gap in the upper bound derived in Theorem 3.

B.2 Metric-derivative bound

We now apply Theorem 2 to the curve traced out on $W_2(\mathcal{Y})$ by perturbing the parameters of the transport map. For a fixed input $x \in \mathcal{X}$ and a direction $h \in \mathbb{R}^p$ in parameter space, we consider the curve

$$t \mapsto \mu_t^h := T_{\theta+th}(x)_{\#}\,\nu. \tag{88}$$

The main theorem below shows that the curve is absolutely continuous in $W_2(\mathcal{Y})$ under mild regularity on T. The metric derivative at $t = 0$ measures the local Wasserstein-rate of change of the predicted distribution under the perturbation h, which we bound by a quadratic form in h involving the parameter sensitivity matrix $G_\theta(x)$.

Theorem 3 (Metric-derivative bound)**.** *Fix $x \in \mathcal{X}$ and $h \in \mathbb{R}^p$. Assume there exists an open neighborhood $U \ni \theta$ such that $(\theta', y) \mapsto T_{\theta'}(x)(y)$ is C^1 on $U \times \mathcal{Y}$, that $T_{\theta'}(x) : \mathcal{Y} \to \mathcal{Y}$ is a C^1 diffeomorphism for every $\theta' \in U$, and that $\int \sup_{\theta'\in U} \|D_\theta T_{\theta'}(x)(y)\|^2_{\mathrm{op}}\,d\nu(y) < \infty$. Then in a neighborhood of $t = 0$, the curve*

$$\mu_t^h := T_{\theta+th}(x)_{\#}\,\nu \tag{89}$$

is absolutely continuous in $W_2(\mathcal{Y})$, and its upper metric derivative at $t = 0$ satisfies

$$|\mu_h'|^+(0)^2 \le h^\top G_\theta(x)\,h, \tag{90}$$

where $G_\theta(x)$ is given by (12). In particular, whenever the metric derivative $|\mu_h'|(0)$ exists, (90) holds also for $|\mu_h'|(0)$.

Proof. We drop the dependence on x throughout. Let $U \ni \theta$ be the open neighborhood from the hypotheses. Choose $\delta > 0$ small enough that $\theta + th \in U$ for all $t \in [-\delta, \delta]$. We define

$$M := \int \sup_{\theta'\in U} \|D_\theta T_{\theta'}(y)\|^2_{\mathrm{op}}\,d\nu(y), \tag{91}$$

which is finite by the integrability assumption.

Consider the candidate vector field

$$v_t^h(y) := D_\theta T_{\theta+th}\big(T^{-1}_{\theta+th}(y)\big)\,h, \qquad t \in (-\delta,\delta). \tag{92}$$

This is well-defined: $T_{\theta+th}$ is a C^1 diffeomorphism for $t \in (-\delta,\delta)$, so $T^{-1}_{\theta+th}(y) \in \mathcal{Y}$ exists for every $y \in \mathcal{Y}$, and $D_\theta T_{\theta+th}$ exists at every point of $\mathcal{Y}$ by joint C^1 regularity on $U \times \mathcal{Y}$. The curve (μ_t^h) is continuous with respect to W_2, since $W_2(\mu_s^h, \mu_t^h) \le \|T_{\theta+sh} - T_{\theta+th}\|_{L^2(\nu)} \to 0$ as $s \to t$ and dominated convergence. We verify the remaining hypotheses of Proposition 5.

Step 1: v_t^h has bounded $L^2(\mu_t^h)$ norm. By definition of the $L^2(\mu_t^h)$ norm and the pushforward identity $\mu_t^h = T_{\theta+th}\#\nu$,

$$\|v_t^h\|^2_{L^2(\mu_t^h)} = \int \left\| D_\theta T_{\theta+th}\big(T^{-1}_{\theta+th}(y)\big)\, h \right\|^2 d\mu_t^h(y) \tag{93}$$

$$= \int \|D_\theta T_{\theta+th}(z)\, h\|^2 \, d\nu(z) \tag{94}$$

$$= h^\top G_{\theta+th}(x)\, h, \tag{95}$$

where the second equality uses the change of variables $z = T^{-1}_{\theta+th}(y)$ (justified by the diffeomorphism assumption), and the third equality applies the definition of G from (12). Bounding the integrand in (94) by $\|h\|^2 \sup_{\theta' \in U} \|D_\theta T_{\theta'}(z)\|^2_{\mathrm{op}}$ and using (91),

$$\|v_t^h\|^2_{L^2(\mu_t^h)} \le \|h\|^2 M. \tag{96}$$

Hence $\int_{-\delta}^{\delta} \|v_t^h\|^2_{L^2(\mu_t^h)}\, dt \le 2\delta \|h\|^2 M < \infty$.

Step 2: v_t^h solves the continuity equation. For any test function $\phi \in C_c^\infty(\mathcal{Y})$, the pushforward identity $\mu_t^h = T_{\theta+th}\#\nu$ gives

$$\frac{d}{dt} \int \phi(y)\, d\mu_t^h(y) = \frac{d}{dt} \int \phi(T_{\theta+th}(z))\, d\nu(z). \tag{97}$$

Since $\nabla\phi$ is bounded, the derivative of the integrand is dominated by $\|\nabla\phi\|_\infty \|h\| \sup_{\theta' \in U} \|D_\theta T_{\theta'}(z)\|_{\mathrm{op}}$, which is ν-integrable by the integrability assumption. By dominated convergence, we interchange d/dt and the integral, and apply the chain rule:

$$\frac{d}{dt} \int \phi(T_{\theta+th}(z))\, d\nu(z) = \int \nabla\phi(T_{\theta+th}(z))^\top D_\theta T_{\theta+th}(z)\, h\, d\nu(z). \tag{98}$$

Changing variables back via $y = T_{\theta+th}(z)$ and using (92):

$$\int \nabla\phi(T_{\theta+th}(z))^\top D_\theta T_{\theta+th}(z)\, h\, d\nu(z) = \int \nabla\phi(y)^\top v_t^h(y)\, d\mu_t^h(y). \tag{99}$$

Integration by parts yields:

$$\int \nabla\phi(y)^\top v_t^h(y)\, d\mu_t^h(y) = -\int \phi(y)\, \nabla \cdot (v_t^h \mu_t^h)(y)\, dy. \tag{100}$$

Combining all the equalities, we have

$$\int \phi\, \partial_t \mu_t^h = -\int \phi\, \nabla \cdot (v_t^h \mu_t^h) \qquad \text{for all } \phi \in C_c^\infty(\mathcal{Y}), \tag{101}$$

which is the weak form of the continuity equation

$$\partial_t \mu_t^h + \nabla \cdot (v_t^h \mu_t^h) = 0 \qquad \text{on } \mathcal{Y} \times (-\delta, \delta). \tag{102}$$

Step 3: Conclusion. By Steps 1 and 2, Proposition 5 applies: $(\mu_t^h)_{t \in (-\delta,\delta)}$ is absolutely continuous in $W_2(\mathcal{Y})$, and

$$|\mu_h'|(r) \le \|v_r^h\|_{L^2(\mu_r^h)} \qquad \text{for a.e. } r \in (-\delta, \delta). \tag{103}$$

Since this bound only holds almost everywhere, we integrate it to obtain the limit superior around zero. By Proposition 4 and (103), for $0 < |s| < \delta$,

$$W_2(\mu_s^h, \mu_0^h) \le \left| \int_0^s |\mu_h'|(r)\, dr \right| \le \left| \int_0^s \|v_r^h\|_{L^2(\mu_r^h)}\, dr \right|. \tag{104}$$

By (95), $\|v_r^h\|_{L^2(\mu_r^h)} = \sqrt{h^\top G_{\theta+rh}(x)\, h}$, which is continuous in r: the integrand in (94) is continuous in r by joint C^1 regularity and dominated by $\|h\|^2 \sup_{\theta' \in U} \|D_\theta T_{\theta'}(z)\|^2_{\mathrm{op}}$, which is integrable by (91). Dividing by $|s|$, the right-hand side converges to $\sqrt{h^\top G_\theta(x)\, h}$ as $s \to 0$. Taking the $\limsup$ of the left-hand side gives

$$|\mu_h'|^+(0)^2 \le h^\top G_\theta(x)\, h, \tag{105}$$

which is the claimed bound (90). Finally, since $|\mu_h'|^+(0) = |\mu_h'|(0)$ whenever the latter exists (Definition 2), the bound also holds for $|\mu_h'|(0)$ in that case.

□

Theorem 3 establishes that the parameter sensitivity matrix $G_\theta(x)$ approximately controls the local Wasserstein-rate of change of the predicted distribution under any parameter perturbation h: along the curve $t \mapsto T_{\theta+th}(x)\#\nu$, the metric derivative at $t = 0$ is bounded by the quadratic form $\sqrt{h^\top G_\theta(x)\, h}$.

This recovers the parameter sensitivity matrix that appears in Theorem 1, but via a route that is purely geometric: rather than passing through a posterior expectation and Taylor expansion, the bound emerges from the velocity field that the parameter perturbation induces on the Wasserstein manifold.

B.3 Gap analysis

Theorem 3 bounds the upper metric derivative $|\mu_h'|^+(0)$ by $\|h\|_{G_\theta}$, but the inequality is strict in general. We characterize the gap in two cases. First, in dimension one with monotone transport maps, the bound is exact: $G_\theta(x)$ recovers the true metric derivative. Second, in higher dimensions the gap is governed by the orthogonal decomposition of the velocity field into a tangent component and a divergence-free component, with the tangent component achieving equality. The two results are summarized in Lemma 3 and Lemma 4, respectively.

B.3.1 Tightness in dimension one

Lemma 3 (1D tightness)**.** *Under the hypotheses of Theorem 3 with* $\dim(\mathcal{Y}) = 1$*, assume further that* ν *is absolutely continuous with respect to the Lebesgue measure, and that* $T_{\theta'}(x)$ *monotone in* y *for every* $\theta' \in U$*. Then, the inequality* (90) *becomes an equality:*

$$|\mu_h'|(t)^2 = h^\top G_{\theta+th}(x)\, h \qquad \text{for all } t \in (-\delta, \delta). \tag{106}$$

Proof. Monotonicity of $T_{\theta+th}(x)$ gives the quantile representation $T_{\theta+th}(x) = F^{-1}_{\theta+th}(x;\cdot) \circ F_\nu$, where $F^{-1}_{\theta+th}(x;\tau)$ is the τ-quantile of $\mu_t^h = T_{\theta+th}(x)\#\nu$.

Step 1: metric derivative. The 1D quantile representation of the squared 2-Wasserstein distance gives, for $s, t \in (-\delta, \delta)$,

$$W_2^2(\mu^h_{t+s}, \mu^h_t) = \int_0^1 \left(F^{-1}_{\theta+(t+s)h}(x;\tau) - F^{-1}_{\theta+th}(x;\tau)\right)^2 d\tau. \tag{107}$$

Dividing by s^2 and passing to the limit $s \to 0$ (interchanging limit and integral by dominated convergence),

$$|\mu_h'|(t)^2 = \int_0^1 \left|\frac{\partial}{\partial t} F^{-1}_{\theta+th}(x;\tau)\right|^2 d\tau. \tag{108}$$

Step 2: L^2 norm of the velocity field. By (95) from the proof of Theorem 3,

$$h^\top G_{\theta+th}(x)\, h = \|v_t^h\|^2_{L^2(\mu_t^h)} = \int \left\|v_t^h(y)\right\|^2 d\mu_t^h(y). \tag{109}$$

The velocity field, defined in (92), satisfies

$$v_t^h(y) = D_\theta T_{\theta+th}(x)(T^{-1}_{\theta+th}(x)(y))\, h = \frac{\partial}{\partial t} T_{\theta+th}(x)(T^{-1}_{\theta+th}(x)(y)), \tag{110}$$

where the second equality applies the chain rule to the parameterization $t \mapsto T_{\theta+th}(x)(z)$ at fixed z. Substituting,

$$\|v_t^h\|^2_{L^2(\mu_t^h)} = \int \left|\frac{\partial}{\partial t} T_{\theta+th}(x)(T^{-1}_{\theta+th}(x)(y))\right|^2 d\mu_t^h(y). \tag{111}$$

Now we use monotonicity to rewrite this in quantile form. Since $T_{\theta+th}(x) = F^{-1}_{\theta+th}(x;\cdot) \circ F_\nu$, we have $T^{-1}_{\theta+th}(x)(y) = F_\nu^{-1}(F_{\theta+th}(x;y))$, where $F_{\theta+th}(x;\cdot)$ is the CDF of μ_t^h. Combining,

$$\frac{\partial}{\partial t} T_{\theta+th}(x)(T^{-1}_{\theta+th}(x)(y)) = \frac{\partial}{\partial t} F^{-1}_{\theta+th}(x; F_{\theta+th}(x;y)). \tag{112}$$

Applying the change of variables $\tau = F_{\theta+th}(x;y)$ (so $d\mu_t^h(y) = d\tau$) in (111),

$$\|v_t^h\|^2_{L^2(\mu_t^h)} = \int_0^1 \left|\frac{\partial}{\partial t} F^{-1}_{\theta+th}(x;\tau)\right|^2 d\tau. \tag{113}$$

Comparing (108) and (113) gives $|\mu_h'|(t)^2 = h^\top G_{\theta+th}(x)\, h$, which is the claimed equality. □

B.3.2 Decomposition of the velocity field in higher dimensions

Let $L^2(\mu_t^h;\mathbb{R}^d)$ denote the Hilbert space of $\mathbb{R}^d$-valued vector fields $v : \mathcal{Y} \to \mathbb{R}^d$ that are square-integrable with respect to μ_t^h, equipped with the inner product $\langle u, v\rangle_{L^2(\mu_t^h)} = \int \langle u(y), v(y)\rangle \, d\mu_t^h(y)$. The tangent space $\mathrm{Tan}_{\mu_t^h}\mathcal{P}_2(\mathcal{Y})$ is a closed subspace of $L^2(\mu_t^h;\mathbb{R}^d)$, and its orthogonal complement consists of the vector fields w that are divergence-free with respect to μ_t^h, i.e., $\nabla \cdot (w\,\mu_t^h) = 0$ in the distributional sense; equivalently, $\int \langle \nabla\phi, w\rangle \, d\mu_t^h = 0$ for every $\phi \in C_c^\infty(\mathcal{Y})$. This is the Helmholtz–Hodge decomposition in $L^2(\mu_t^h)$ [68, 69].

We may therefore consider the orthogonal projection operator

$$\Pi : L^2(\mu_t^h;\mathbb{R}^d) \to \mathrm{Tan}_{\mu_t^h}\mathcal{P}_2(\mathcal{Y}), \tag{114}$$

that sends each vector field to its component along $\mathrm{Tan}_{\mu_t^h}\mathcal{P}_2(\mathcal{Y})$. Given that we characterized the tangent space as the closure of gradient fields (Definition 3), the projection $\Pi(v)$ returns the best $L^2(\mu_t^h)$-approximation of v by such fields. The next lemma shows that this projection characterizes the gap in Theorem 3.

Lemma 4 (Gap as the divergence-free component)**.** *Let v_t^h be the velocity field defined in (92), and write*

$$v_t^h = v_t^{\tan} + w_t, \qquad v_t^{\tan} := \Pi(v_t^h), \qquad w_t := v_t^h - v_t^{\tan}, \tag{115}$$

so that $v_t^{\tan} \in \mathrm{Tan}_{\mu_t^h}\mathcal{P}_2(\mathcal{Y})$ and w_t is divergence-free with respect to μ_t^h. Then the gap in (90) satisfies

$$h^\top G_{\theta+th}(x)h = \|w_t\|^2_{L^2(\mu_t^h)} + |\mu_h'|(t)^2, \qquad a.e.\ t. \tag{116}$$

In particular, equality in (90) holds if and only if $w_t = 0$, i.e., the velocity field v_t^h lies in the tangent space.

Proof. By orthogonality of the decomposition $v_t^h = v_t^{\tan} + w_t$ in $L^2(\mu_t^h)$,

$$\|v_t^h\|^2_{L^2(\mu_t^h)} = \|v_t^{\tan}\|^2_{L^2(\mu_t^h)} + \|w_t\|^2_{L^2(\mu_t^h)}. \tag{117}$$

$v_t^{\tan}$ still solves the continuity equation, as $\nabla \cdot (\mu_t w_t) = 0$. By Theorem 6, the metric derivative coincides with the $L^2(\mu_t^h)$-norm of the tangent component:

$$|\mu_h'|(t)^2 = \|v_t^{\tan}\|^2_{L^2(\mu_t^h)}. \tag{118}$$

Combining (117) and (118), and using $\|v_t^h\|^2_{L^2(\mu_t^h)} = h^\top G_{\theta+th}(x)\, h$ from (95),

$$h^\top G_{\theta+th}(x)\, h = |\mu_h'|(t)^2 + \|w_t\|^2_{L^2(\mu_t^h)}. \tag{119}$$

The "if and only if" claim follows since $\|w_t\|^2 \geq 0$ with equality iff $w_t = 0$. □

Lemma 4 suggests a refinement of $G_\theta(x)$: replacing v_0^h with its tangent projection $\Pi(v_0^h)$ gives a quadratic form $h^\top \tilde{G}_\theta(x)\, h := \|\Pi(v_0^h)\|^2_{L^2(\mu_0^h)}$ that exactly recovers the squared metric derivative $|\mu_h'|(0)^2$. We leave this as a future research direction.

C Exact metric tensor under the Knothe–Rosenblatt distance

Theorem 3 bounds the metric derivative on the Wasserstein-2 manifold of the probability curve $t \mapsto \mu_{\theta+th}(x)$, evaluated at $t = 0$ by the quadratic form $h^\top G_\theta(x)h$. We now show that for lower-triangular affine transport maps (e.g., obtained via Cholesky parameterization), this bound is exact under the Knothe–Rosenblatt (KR) distance [70, 71]. In this setting, $G_\theta(x)$ is the exact metric tensor, suggesting that the parameter sensitivity matrix has a precise geometric meaning beyond the loose Wasserstein-2 upper bound.

C.1 Background: KR rearrangement, KR distance, and metric derivative

Definition 4 (Knothe–Rosenblatt rearrangement)**.** *Let ν, μ be probability measures on $\mathbb{R}^d$. A* Knothe–Rosenblatt rearrangement *from ν to μ is a triangular map $T^{\mathrm{KR}} : \mathbb{R}^d \to \mathbb{R}^d$ that is component-wise monotone in its arguments and satisfies $T^{\mathrm{KR}}_{\#}\nu = \mu$.*

When ν is absolutely continuous with respect to the Lebesgue measure, the KR rearrangement from ν to any target μ exists and is ν-a.e. unique [70, 71].

Definition 5 (Knothe–Rosenblatt distance)**.** *Let ν be a probability measure on $\mathbb{R}^d$ absolutely continuous with respect to the Lebesgue measure, and let μ_1, μ_2 be probability measures on $\mathbb{R}^d$ with finite second moments. Let $T_1^{\mathrm{KR}}, T_2^{\mathrm{KR}}$ be the KR rearrangements from ν to μ_1, μ_2 respectively. The squared* Knothe–Rosenblatt distance *between μ_1 and μ_2 is*

$$W^2_{\mathrm{KR}}(\mu_1, \mu_2) := \int \|T_1^{\mathrm{KR}}(y) - T_2^{\mathrm{KR}}(y)\|_2^2 \, d\nu(y). \tag{120}$$

Definition 6 (Metric derivative under W_{KR})**.** *For the curve $t \mapsto \mu_\theta^t$ along a perturbation direction $h \in \mathbb{R}^p$, the KR metric derivative at $t = 0$ is*

$$|\mu_h'|_{\mathrm{KR}}(0) := \lim_{t\to 0} \frac{W_{\mathrm{KR}}(\mu_\theta^t, \mu_\theta^0)}{|t|}, \tag{121}$$

when the limit exists.

C.2 KR metric derivative for lower-triangular affine pushforwards

We first specialize the KR distance to the family of lower-triangular affine pushforwards relevant to our setting.

Lemma 5 (KR distance for lower-triangular affine pushforwards)**.** *Let ν be a probability measure on $\mathbb{R}^d$ absolutely continuous with respect to the Lebesgue measure, with finite second moments. Let T_1, T_2 be lower-triangular affine maps $T_i(y) = b_i + L_i y$ with L_i having strictly positive diagonals. Then T_i is the KR rearrangement from ν to $T_{i\#}\nu$, and*

$$W^2_{\mathrm{KR}}(T_{1\#}\nu, T_{2\#}\nu) = \int \|T_1(y) - T_2(y)\|_2^2 \, d\nu(y). \tag{122}$$

Proof. By the discussion following Definition 4, T_i is triangular and component-wise monotone (since L_i has positive diagonal entries), hence T_i is a KR rearrangement from ν to $T_{i\#}\nu$. By the ν-a.e. uniqueness of the KR rearrangement under the absolute continuity assumption on ν, T_i is the unique such map. Substituting into Definition 5 gives the claimed formula. □

We now show that in contrast to the Wasserstein-2 bound (Theorem 3), G_θ exactly captures the squared metric derivative under the KR distance for lower-triangular affine pushforwards.

Theorem 4 (Exact metric tensor for lower-triangular affine maps)**.** *Let $T_\theta(y) = b_\theta + L_\theta y$ be a lower-triangular affine transport map with L_θ having strictly positive diagonals, and assume $\theta \mapsto (b_\theta, L_\theta)$ is differentiable. Assume an absolutely continuous measure ν has zero mean and identity covariance. Then for any perturbation direction $h \in \mathbb{R}^p$,*

$$|\mu'(\theta, h)|^2_{\mathrm{KR}} = h^\top G_\theta \, h, \quad \textit{where} \quad G_\theta = \int D_\theta T_\theta(y)^\top D_\theta T_\theta(y) \, d\nu(y). \tag{123}$$

Proof. By Lemma 5, for any two affine maps $T_i(y) = b_i + L_i y$:

$$W_{\mathrm{KR}}^2(T_{1\#}\nu, T_{2\#}\nu) = \int \|T_1(y) - T_2(y)\|^2 \, d\nu(y). \tag{124}$$

Expanding $T_1(y) - T_2(y) = (b_1 - b_2) + (L_1 - L_2)y$ and using $\mathbb{E}_\nu[y] = 0$ and $\mathbb{E}_\nu[yy^\top] = I$, the cross-term vanishes and the integral evaluates to

$$W_{\mathrm{KR}}^2(T_{1\#}\nu, T_{2\#}\nu) = \|b_1 - b_2\|^2 + \|L_1 - L_2\|_F^2. \tag{125}$$

Applying (125) to the parameter curve $T_{\theta+th}$ vs. T_θ,

$$W_{\mathrm{KR}}^2(\mu(\theta + th), \mu(\theta)) = \|b_{\theta+th} - b_\theta\|^2 + \|L_{\theta+th} - L_\theta\|_F^2. \tag{126}$$

A first-order Taylor expansion of b_θ and L_θ in θ gives, as $t \to 0$,

$$\|b_{\theta+th} - b_\theta\|^2 = t^2 \|D_\theta b_\theta \, h\|^2 + o(t^2), \tag{127}$$

$$\|L_{\theta+th} - L_\theta\|_F^2 = t^2 \|D_\theta L_\theta \, h\|_F^2 + o(t^2). \tag{128}$$

Dividing by t^2 and taking the limit,

$$|\mu'(\theta, h)|_{\mathrm{KR}}^2 = \|D_\theta b_\theta \, h\|^2 + \|D_\theta L_\theta \, h\|_F^2. \tag{129}$$

By Lemma 2, this equals $h^\top G_\theta h$ where $G_\theta = \int D_\theta T_\theta(y)^\top D_\theta T_\theta(y) \, d\nu(y)$. □

D Experimental details

D.1 Conditional Histogram Experimental Details

D.1.1 Data Simulation

The input space is $\mathcal{X} = [0, 1]$. For each x, the conditional distribution μ_x is a symmetric two-component Gaussian mixture

$$\mu_x = \frac{1}{2}\mathcal{N}\left(m(x) - d(x),\ \sigma^2(x)\right) + \frac{1}{2}\mathcal{N}\left(m(x) + d(x),\ \sigma^2(x)\right), \tag{130}$$

with conditional mean $m(x) = x$, mode half-separation $d(x) = \exp\left(-20\,(x - 0.5)^2\right)$, and component standard deviation $\sigma(x) = 0.2\,(1 + x)^2$.

This design is chosen so that $m(x)$ is smooth while the quantile functions are localized. With this setting, we anticipate that a mean-based active learning method will sample the space evenly while a quantile-based active learning method will be sensitive to areas of $\mathcal{X}$ where the variance changes.

D.1.2 Gaussian Process Regression

We fit a quantile function $x \mapsto F_x^{-1}(\tau_j)$ with $F_x^{-1}(\tau_j) = m_\theta(x) + g_\theta(x; \tau_j)$ for each quantile $\tau_j \in \{\tau_1, \ldots, \tau_M\}$ on a uniform grid ($M = 30$, splitting $[0.03, 0.97]$ for stability, boundaries included) independently with a Gaussian process. We estimate posterior expected variance in (40) by integrating the variance across the quantiles. For the mean strategy, we fit a Gaussian process predicting the mean and obtain the posterior expected variance. Kernel hyperparameters (amplitude and bandwidth of an RBF kernel) are pre-fitted by maximum likelihood on a dense ground-truth grid and then fixed throughout the active learning loop; this isolates the effect of the acquisition strategy from hyperparameter estimation artifacts.

D.1.3 Active Learning Strategies

Wasserstein AL Select

$$x_{\text{new}} = \arg\min_{x' \in \mathcal{X}_{\text{cand}}} \mathbb{E}_{x^* \sim p_{\text{test}}} \int_0^1 \operatorname{Var}\left[F_{x^*}^{-1}(\tau) \mid \mathcal{D} \cup \{x'\}\right] d\tau, \tag{131}$$

targeting the full distributional risk.

Mean AL Select

$$x_{\text{new}} = \arg\min_{x' \in \mathcal{X}_{\text{cand}}} \mathbb{E}_{x^* \sim p_{\text{test}}} \operatorname{Var}[m(x^*) \mid \mathcal{D} \cup \{x'\}], \tag{132}$$

targeting only the first term (mean) of the risk decomposition.

Random Select x_{new} uniformly at random.

D.1.4 Experiments

Each strategy starts from $N_{\text{init}} = 5$ shared initial observations (including the boundary points $x = 0.005$ and $x = 0.995$) and acquires 5 additional points sequentially.

Mean Wasserstein loss (35), mean squared error between the predicted mean $m_\theta(x) = \int F_\theta^{-1}(x) d\tau$ and the true mean, and the mean shape error (37) across conditions are reported.

D.2 Baselines

The NTK baseline benchmarked throughout Section 5 is the standard finite-width NTK (4) for last-layer parameters, applied to the mean-fitting network that is trained to predict the sample mean. The size of the mean-fitting network is set as identical to the transport map trained. The NTK kernels are paired with the same acquisition rule (18) as tNTK.

For the conditional quantile prediction, we have two additional baselines that use NTK of networks that each predict mean and variance, and quartiles. We also apply BALSA^{EMD}from Werner and

Schmidt-Thieme [56], which is the only work that takes the distribution into account to the best of our knowledge. BALSA^{EMD} was benchmarked only for conditional quantile prediction, as it is only designed for 1-dimensional distributions.

D.3 Affine transport of 2D Gaussian

In this section, we derive a fully analytic expression for tNTK when the conditional generative model is a two-dimensional affine map whose covariance factor is parameterized via a Cholesky decomposition, along with the experimental details for the results presented in Section 5.1. The resulting kernel requires no Monte Carlo integration over the base measure and no automatic differentiation—only inner products of intermediate features and pointwise sigmoid evaluations. We then describe the details about training and data.

D.3.1 Affine Cholesky Parameterization

We first show the general result on the affine map with Cholesky decomposition on a d-dimensional Gaussian, with experimental details deferred to Appendix D.3.2. Let $x \in \mathcal{X}$ denote the conditioning variable and $y = (y_1, \dots, y_d)^\top \sim \nu$ a sample from the base measure $\nu = \mathcal{N}(0, I_d)$. We consider a conditional transport map of the form

$$T_\theta(x)(y) \;=\; b_\theta(x) \;+\; L_\theta(x)\, y \;\in\; \mathbb{R}^d, \tag{133}$$

where $b_\theta(x) \in \mathbb{R}^d$ is the predicted mean and $L_\theta(x) \in \mathbb{R}^{d\times d}$ is a lower-triangular matrix with strictly positive diagonal entries, so that $\Sigma_\theta(x) = L_\theta(x)\, L_\theta(x)^\top$ is a valid covariance matrix. The pushforward distribution is $\mu(\theta, x) = \mathcal{N}\big(b_\theta(x),\, \Sigma_\theta(x)\big)$.

Both $b_\theta(x)$ and $L_\theta(x)$ are produced by a neural network with the following last-layer structure.

Backbones. Two feature extractors (which may or may not share parameters) map the input to representations

$$h_\theta(x) \;\in\; \mathbb{R}^H, \qquad g_\theta(x) \;\in\; \mathbb{R}^H, \tag{134}$$

where h_θ feeds the mean head and g_θ feeds the Cholesky head. Their parameters are part of θ but are held fixed when defining the *last-layer* kernel.

Mean head. A linear layer with weight rows $m_j^\top, \in \mathbb{R}^H,\ j = 1, \dots, d$ produces the mean vector:

$$b_\theta^j(x) \;=\; m_j^\top h_\theta(x) + b_j. \tag{135}$$

Cholesky head. A linear layer with weight rows $\ell_{ji}^\top \in \mathbb{R}^H$ and biases $k_{ji} \in \mathbb{R}$ for $j = 1, \dots, d,\ i = 1, \dots, j$ produces the elements of the lower-triangular factor, where the softplus function $\sigma(t) = \log(1 + e^t)$ is applied to the diagonal entries ($i = j$).

$$L_\theta(x) \;=\; \begin{bmatrix} \sigma\big(\ell_{11}^\top g_\theta(x) + k_{11}\big) & 0 & \dots & 0 \\ \ell_{21}^\top g_\theta(x) + k_{21} & \sigma\big(\ell_{22}^\top g_\theta(x) + k_{22}\big) & \dots & 0 \\ \vdots & & & \vdots \\ \ell_{d1}^\top g_\theta(x) + k_{d1} & \dots & & \sigma\big(\ell_{dd}^\top g_\theta(x) + k_{dd}\big) \end{bmatrix}. \tag{136}$$

Last-layer gradients. Conditional transport can be written as

$$T_\theta^j(x)(y) = m_j^\top h(x) + b_j + \sum_{i=1}^{j-1} (\ell_{ji}^\top g_\theta(x) + k_{ji}) y_i + \sigma\left(l_{jj}^\top g_\theta(x) + k_{jj}\right) y_j \tag{137}$$

where $T_\theta^j(x)(y)$ denotes j-th dimension of the output. We compute the gradient of $T_\theta^j(x)(y)$ with respect to the last-layer parameters $\theta = \{m, b, \ell, k\}$. Let σ' denote the sigmoid function (the derivative

of softplus).

$$\nabla_{m_j} T_\theta^j(x)(y) = h_\theta(x) \tag{138}$$

$$\nabla_{b_j} T_\theta^j(x)(y) = 1 \tag{139}$$

$$\nabla_{\ell_{ji}} T_\theta^j(x)(y) = g_\theta(x) y_i, \quad i = 1, \ldots, j-1 \tag{140}$$

$$\nabla_{k_{ji}} T_\theta^j(x)(y) = y_i, \quad i = 1, \ldots, j-1 \tag{141}$$

$$\nabla_{\ell_{jj}} T_\theta^j(x)(y) = \sigma'\left(\ell_{jj}^\top g_\theta(x) + k_{jj}\right) y_j g_\theta(x) \tag{142}$$

$$\nabla_{k_{jj}} T_\theta^j(x)(y) = \sigma'\left(\ell_{jj}^\top g_\theta(x) + k_{jj}\right) y_j \tag{143}$$

$$\tag{144}$$

and we have the closed-form tNTK:

$$k_\theta(x, x') = \int \mathrm{Tr}\big[D_w T_\theta(x)(y)^\top D_w T_\theta(x')(y)\big]\, d\nu(y), \tag{145}$$

$$= \int \sum_{j=1}^{d} \Bigg(h_\theta(x)^\top h_\theta(x') + 1 + \sum_{i=1}^{j-1} \left(g_\theta(x)^\top g_\theta(x') + 1\right) y_i^2 + \sigma'(\ell_{jj}^\top g_\theta(x) + k_{jj}) \sigma'(\ell_{jj}^\top g_\theta(x') + k_{jj}) \left(g_\theta(x)^\top g_\theta(x') + 1\right) y_j^2 \Bigg) d\nu(y) \tag{146}$$

$$= d\left(h_\theta^\top h_\theta(x') + 1\right) + \tag{147}$$

$$\left(\left(\frac{d(d-1)}{2} + \sum_j \sigma'(\ell_{jj}^\top g_\theta(x) + k_{jj}) \sigma'(\ell_{jj}^\top g_\theta(x') + k_{jj}) \right) (g_\theta(x)^\top g_\theta(x') + 1) \right) \tag{148}$$

D.3.2 Training

We use a training pool of 200 conditions and 100 test conditions, with active learning batches of size $B = 2$ over 5 rounds starting from $B_0 = 5$ initial conditions. The mean and covariance heads are separate MLPs, each with 2 hidden layers and ReLU activations followed by a linear output layer of dimension 2 (mean) and 3 (Cholesky factor entries), respectively. We train both networks jointly using the analytical KL divergence between the predicted and ground-truth Gaussian as the loss:

$$D_{\mathrm{KL}}(\mathcal{N}(\mu_t, \Sigma_t) \,\|\, \mathcal{N}(\mu_p, \Sigma_p)) = \frac{1}{2}\left[\mathrm{tr}(\Sigma_p^{-1}\Sigma_t) + (\mu_p - \mu_t)^\top \Sigma_p^{-1}(\mu_p - \mu_t) - d + \log \frac{\det \Sigma_p}{\det \Sigma_t}\right] \tag{149}$$

$$= \frac{1}{2}\left[\|L_p^{-1} L_t\|_F^2 + \|L_p^{-1}(\mu_p - \mu_t)\|_2^2 - d + 2\sum_{i=1}^{d} \log(L_p)_{ii} - 2\sum_{i=1}^{d} \log(L_t)_{ii}\right]. \tag{150}$$

The loss is optimized with Adam using a learning rate of 0.001 for 2000 steps. For the evaluation, the Wasserstein-2 distance between predicted and ground truth Gaussian distributions was calculated with its closed form [72].

D.3.3 Data

Data is generated with condition covariates $x = (x_1, x_2)$ that specifies $b_\theta(x)$ and $\Sigma_\theta(x)$:

$$b_\theta(x) = 2.0[e^{-6x_1x_2}, e^{-6x_1(1-x_2)}]^\top \tag{151}$$

$$\Sigma_\theta(x) = \begin{bmatrix} \lambda_1^2 \cos^2\varphi + \lambda_2^2 \sin^2\varphi & (\lambda_1^2 - \lambda_2^2)\sin\varphi\cos\varphi \\ (\lambda_1^2 - \lambda_2^2)\sin\varphi\cos\varphi & \lambda_1^2 \sin^2\varphi + \lambda_2^2 \cos^2\varphi \end{bmatrix} \tag{152}$$

with $\varphi = \pi x_2$, $\lambda_1 = 0.2 + 2e^{-10(1-x_1)}$, and $\lambda_2 = 0.2$. We construct 196 training conditions on a 14×14 Cartesian grid in $[0, 1]^2$, and 100 test conditions on a 10×10 grid uniformly spaced in $[1/28,\, 1 - 1/28]^2$ to avoid overlap with the training grid. The Cholesky factor $L_\theta(x)$ is computed from $\Sigma_\theta(x)$.

D.4 Conditional quantile prediction

D.4.1 Model and training

We train MLPs that produce 32 quantile values per condition following Moon et al. [54]. The MLP outputs a vector $z \in \mathbb{R}^H$ from its last hidden layer. Two linear heads then produce: (i) the smallest predicted quantile value $\hat{F}_\theta^{-1}(\tau_1) = h_{\min}^\top z + b_{\min} \in \mathbb{R}$, and (ii) 31 non-negative increments $\delta = \mathrm{softplus}(Wz + b) \in \mathbb{R}^{31}_{\geq 0}$. The full vector of 32 predicted quantiles is then

$$\hat{F}_\theta^{-1}(\tau_k) = \hat{F}_\theta^{-1}(\tau_1) + \sum_{j=1}^{k-1} \delta_j, \quad k = 1, \ldots, 32, \tag{153}$$

ensuring monotonicity by construction (*non-crossing quantiles*). The targeted quantile levels are $\tau_k = (2k-1)/64$ for $k = 1, \ldots, 32$, i.e., 32 uniformly spaced points in $[1/64, 1 - 1/64]$. Empirical quantiles at these levels are computed from the per-condition samples and used as training targets.

The model is trained with squared Wasserstein-2 loss with the Adam optimizer for 200 minimum and 3000 maximum epochs with early stopping patience of 10 with validation loss. Validation loss is calculated as the loss with empirical quantile values calculated from 20% of held-out per-condition validation samples.

For each dataset, we select the model configuration and weight decay by 5-fold cross-validation over 10 conditions, choosing the combination that minimizes the squared Wasserstein-2 distance on held-out conditions. The search grid is:

- Model configurations $(\text{width}, \text{depth})$: $(64, 2)$, $(32, 3)$, $(128, 3)$.
- Weight decay: $\{0.1, 0.01, 10^{-3}, 10^{-4}\}$.

The selected hyperparameters are reported in Table S1.

Table S1: Selected hyperparameters for the conditional quantile network, per dataset.

Dataset	B_0	$(\text{width}, \text{depth})$	# bins	Weight decay
Simulated Gaussian mixture	5	$(128, 3)$	32	10^{-4}
	10	$(32, 3)$	32	10^{-3}
Temperature	5	$(64, 2)$	32	10^{-3}
	10	$(32, 3)$	32	10^{-2}
Dew Point	5	$(64, 2)$	32	10^{-4}
	10	$(64, 2)$	32	10^{-1}

Active learning loops of iterations 0-5 over 5 folds are run on CPU with 10 GB memory within 2 hours on an internal cluster.

D.4.2 Data

Simulated 1D mixture Gaussian We simulate 1000 1-dimensional Gaussian mixture distribution with 3 components, where parameters are randomly selected with $\mu_i \sim \mathrm{Unif}(-5, 5), \sigma_i \sim \mathrm{Unif}(1.0, 1.5)$, $i \in (1, 2, 3)$ and $\pi \sim \mathrm{Dirichlet}(0.25, 0.25, 0.25)$. 600 samples are generated from each distribution. Condition embedding of $(\mu_1, \sigma_1, \pi_1, \ldots, \mu_3, \sigma_3, \pi_3)$ is generated by ordering components so that $\mu_1 \leq \mu_2 \leq \mu_3$ for identifiability.

Temperature We take the daily surface temperature measurements between 2023 and 2025 measured across 11396 stations from NOAA National Centers for Environmental Information [55](CC0-1.0 license). 1000 stations with no less than 256 days of measurements, with non-NaN station locations, and with the highest Wasserstein distance from the control distribution of a random Gaussian with mean and standard deviation of all measurements were selected. The distance is calculated from 256 random samples from each station and 1000 random samples from the control distribution. The resulting station had 967-971 days of measurements. $(\text{longitude}, \text{latitude}, \text{elevation})$ was used as 3-dimensional condition embedding for each station. Yearly temperature and each of the condition embedding dimensions were globally scaled to unit variance and 0 mean before training.

Dew point We take the daily dew point measurements from the same data source as in the Temperature dataset. 1000 stations with no less than 512 days of measurements and with non-NaN station locations were randomly selected, resulting in 515-970 days of measurements. The same condition embedding and data scaling were used as in the Temperature dataset.

D.5 Conditional flow matching

D.5.1 Model and training

We adapted the source code of Klein et al. [37] [2] with linear interpolant:

$$\phi^{0,x}(y_i) = y \sim \mathcal{N}(0, I) \tag{154}$$

$$u_i^{t,x}(\phi^{t,x}(y_i)) = \frac{d}{dt}\phi^{t,x}(y_i) = \frac{y_i^x - \phi^{t,x}(y_i)}{1-t} \tag{155}$$

where y_i^x is a sample of condition x. We learn $v_\theta(t, x, \phi^{t,x}(y_i))$ to approximate $u_i^{t,x}(\phi^{t,x}(y_i))$ [32]. The flow ODE is parameterised by a velocity field

$$v_\theta(t, x, \phi^{t,x}(y_i)) = g_\theta\Big(\text{concat}\big(f_\theta^{\text{time}}(t),\ f_\theta^{\text{sample}}(\phi^{t,x}(y_i)),\ f_\theta^{\text{cond}}(x),\big)\Big), \tag{156}$$

where f_θ^{sample}, f_θ^{cond}, and g_θ are MLPs with SiLU activation [73], f_θ^{time} is a Fourier-feature time encoder, and the condition embedding is mean-pooled before f_θ^{cond}. A combination of dropout rates among $(0.0, 0.5)$ and one of three model configurations in Table S2 for each dataset is selected to be the one with the least Sinkhorn divergence in validation conditions from the 5-fold CV of 10 randomly selected conditions.

Table S2: CellFlow model configurations

Config	f_θ^{sample}	f_θ^{cond}	g_θ	Time frequencies	f_θ^{time}
1	$[64, 64]$	$[64, 64]$	$[128, 128, 128, 32]$	32	[128, 32]
2	$[64, 64]$	$[64, 64]$	$[128, 128, 128, 128]$	128	[128, 128]
3	$[256, 256]$	$[64, 64]$	$[512, 512, 512, 64]$	64	[512, 64]

For a given set of training conditions, we hold out 20% of samples per condition as validation samples. At each training step, a 256-cell batch is sampled per condition across 128 time samples of t; one optimization step processes all training conditions jointly. The model is trained with Adam (per-dataset learning rate, Table S3) for up to n_{eval} steps; validation is run every 1000 steps on Sinkhorn divergence ($\epsilon = 0.1\sigma$, where σ is the standard deviation of cost matrix entries) over the held-out validation samples. We additionally apply a plateau-based learning-rate scheduler (patience 5, factor 0.5, $\text{lr}_{\min} = 0.1\text{lr}$) to CLINC 150 dataset and early stopping with validation loss. Model parameters from the best validation step are restored when early stopping is triggered before kernel computation and test evaluation.

Table S3: Per-dataset CellFlow hyperparameters

Dataset	lr	Config	dropout rate	n_{eval}	early-stopping patience
Radial flow	5×10^{-6}	1	0	30,000	5
Van der Pol oscillation	5×10^{-5}	1	0	30,000	5
CLINC 150	5×10^{-5}	1	0.5	300,000	30

Active learning uses 5-fold cross-validation over conditions: each fold holds out $1/5$ of conditions for test evaluation, and the remaining train/validation conditions are revealed in batches across 3 AL iterations. Iteration 0 uses random initial conditions and is therefore strategy-independent; we train it once per (fold, seed) to eliminate GPU non-determinism drift across strategies.

Sinkhorn divergence for validation samples and test conditions is evaluated by integrating the flow ODE with adaptive Tsit5(rtol = atol = 10^{-5}) from 256 standard-normal source samples (sampled

[2] https://github.com/theislab/CellFlow, v0.0.6, MIT license

with replacement if the number of available samples per condition is less than 256) to $t = 1$, with $\epsilon = 0.05\sigma$, where σ is the standard deviation of cost matrix entries.

For the kernel calculation, $M = 4096$ random samples from the standard normal distribution are integrated with the flow ODE with adaptive Tsit5(rtol = atol = 10^{-5}). Active learning loops of iterations 0-3 over 5 folds are run on L40S GPUs with 30GB of memory within 48 hours per seed on an internal cluster.

D.5.2 Data

Simulated radial flow We simulated radial flow in $\mathbb{R}^2$ with a global attractor of concentric circles centered at the origin with radius $\sqrt{\alpha/\beta}$, and the steepness of the attraction is controlled by α, where $y \sim \mathcal{N}(0, I)$.

$$\frac{d}{dt}\phi^{t,x}(y) = (\alpha - \beta\|\phi^{t,x}(y)\|^2)\phi^{t,x}(y), \quad \alpha, \beta > 0 \tag{157}$$

The pool comprises 100 conditions: 50 sampled with jitter around a nominal point, $\alpha \sim \text{Unif}(0.98, 1.02)$ and $\beta \sim \text{Unif}(2.48, 2.52)$, and 50 from the Cartesian product of $\alpha \in \{0.1, 0.31, 0.52, \ldots, 2.0\}$ (10 uniformly spaced points in $[0.1, 2.0]$) and $\beta \in \{0.5, 1.625, 2.75, 3.875, 5.0\}$ (5 uniformly spaced points in $[0.5, 5.0]$). For each condition, we generate 1000 particle trajectories by Euler integration of the radial flow ODE for 100 steps with $\Delta t = 0.01$, yielding the empirical distribution at $t = 1$. The conditioning input is $(\alpha, \beta) \in \mathbb{R}^2$.

Simulated Van der Pol oscillation We simulated Van der Pol oscillation for varying starting particle locations $y \sim \mathcal{N}([c_1, c_2]^\top, \sigma^2 I)$ and parameter μ.

$$\frac{dx_1}{dt} = x_2 \tag{158}$$

$$\frac{dx_2}{dt} = \mu(1 - x_1^2)\, x_2 - x_1 \tag{159}$$

Random direction and radius in the 4-sphere were sampled, where $r = \min(\text{Exp}(2.5), 1)$, Exp denotes samples from exponential distribution, u is the uniformly randomly selected unit vector in 4-sphere, and h is the vector of half ranges for each parameter.

$$\text{Parameter:} \quad \theta = (\mu, c_1, c_2, \sigma) = \text{clip}(\theta_0 + r(u \odot h), \theta_{\min}, \theta_{\max}) \tag{160}$$

$$\theta_0 = (\mu_0, c_{1,0}, c_{2,0}, \sigma) = (2.0, 0.0, 0.0, 0.5) \tag{161}$$

$$\theta_{\min} = (-0.5, -1.5, -1.5, 0.2) \tag{162}$$

$$\theta_{\max} = (3.0, 1.5, 1.5, 1.0) \tag{163}$$

$$h_i = \max(\theta_{0,i} - \theta_{\min,i}, \theta_{\max,i} - \theta_{0,i}) \tag{164}$$

$\phi^{3,x}(y)$ for 1000 particles per condition is calculated with Euler integration with 300 steps ($\Delta t = 0.01$). θ was used as the condition embedding.

CLINC 150 CLINC 150 is an intent classification dataset with 150 intentions and 100 sentences per intention [59, 60] (CC BY 4.0 license). We embedded sentences with the BART [62] base-sized pretrained model, then embedded them into an 8-dimensional PC embedding. The label for each intent class was embedded with the fine-tuned MINILM [61][3] text encoder into 64 latent dimensions with `truncate_dim=64`, then scaled to have unit variance per dimension.

D.6 Conditional Monge gap

D.6.1 Model and training

We adapted the source code of Driessen et al. [31][4]. For each train condition, 1/5 of the samples are held out as validation samples. We train the model for 100 epochs, with one condition per batch, selecting 256 random samples for the condition with replacement. The network was trained with the

[3] all-MiniLM-L6-v2, https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2 under Apache-2.0 license

[4] Source code at https://github.com/AI4SCR/conditional-monge-gap under MIT license, accessed Nov 1st, 2025

Adam optimizer with weight decay of 10^{-5}. Monge gap regularization parameter was set to 1.0. OT regularization parameter was set to $\epsilon\sigma$ with $\epsilon = 0.05$ for Sinkhorn divergence and $\epsilon = 0.1$ for Monge gap, where σ is the standard deviation of all values in the cost matrix for OT matching. The learning rate was decayed with a cosine decay schedule with constant decay per step until 50 epochs, down to $1/10$ of the initial learning rate. Parameters of the epoch with the least validation loss were selected for test condition evaluation and kernel calculation.

The model separately encodes condition and sample, and concatenated results are fed to a shared MLP to predict the transported particle:

$$T_\theta(x, y) = g_\theta(\text{concat}(f_\theta^{\text{cond}}(x), f_\theta^{\text{sample}}(y))) \tag{165}$$

where f_θ^{cond}, f_θ^{sample}, g_θ are MLPs with GELU activation layer [74] and f_θ^{cond} and f_θ^{sample} has last layer activation.

Model configuration and learning rate combination are selected based on the lowest Sinkhorn divergence on the validation conditions from the 5-fold CV of n conditions, where $n = 10$ is the minimum initial batch size. Hyperparameter sweep was across 3 model configurations and 3 learning rates $(0.01, 0.001, 0.0001)$.

Table S4: Conditional Monge Gap model configurations

Config	$f_\theta^{\texttt{sample}}$	$f_\theta^{\texttt{cond}}$	g_θ
1	$[64, 64]$	$[64, 64]$	$[128, 128, 128, 32]$
2	$[128, 128]$	$[128, 128]$	$[256, 256, 256, 64]$
3	$[256, 256]$	$[256, 256]$	$[512, 512, 512, 64]$

Table S5: Selected model configurations for Conditional Monge map datasets

Dataset	Model configuration	Learning rate
K562-Chemical	3	0.0001
BMDM-Genetic	3	0.001

Test conditions are evaluated with 3 random samples with replacement of 256 samples per condition, both from the source and target samples.

$M = 4096$ samples were used to obtain Monte Carlo estimates of kernels. Active learning loops of iterations 0-5 over 5 folds are run on CPU with 100GB of memory within 48 hours on an internal cluster.

D.6.2 Datasets

K562-Chemical K562 cells and MCF7 cells from the Sciplex3 dataset [75, 64] (CC BY 4.0 license) are separately embedded into scVI [76] 10-dimensional embeddings with default parameters, and then the mean response embedding in MCF7 is used as the condition covariates. 436 kinase-dose conditions are selected that have at least 80 cells profiled in K562 and are present in MCF7, yielding 64,971 cells, with 3,359 unperturbed cells (80-255 per condition; median 138).

BMDM-Genetic We take the subset of Perturb-seq outcomes that targeted 514 genes in mouse bone marrow-derived macrophages (BMDMs) from Kudo et al. [77]. 3000 highly variable genes are selected, and a 50-dimensional salient representation [78] is obtained. The imaging outcome of the optical pooled screen [79] with matched perturbations was processed and embedded into 70 principal components and pseudo-bulked per perturbation. The top 198 perturbations that diverge the most from the control cells with the largest Wasserstein distance in the 50-dimensional latent space are selected for the experiment in this paper, leading to 73,161 cells with 25,952 unperturbed cells and 256-1,963 cells for each of the 198 selected perturbations. The response of all cells, including the

unperturbed cells, was projected onto the first 10-dimensional principal components. Further details of the dataset and preprocessing are described in the original manuscript Kudo et al. [77][5].

[5]Public repository appears at https://github.com/Genentech/MoCAVI/blob/main/demos/tutorial_mocavi_kinase_screen.ipynb

E Additional Results

E.1 Empirical analysis of gap in Theorem 1

We assessed how loose the upper bound becomes as the dimension grows by analyzing the transport of d-dimensional Gaussian distributions. We compute the analytical kernel as described in Appendix D.3. We compute the Fréchet variance by obtaining the barycenter of the posterior distribution with randomly sampled parameters $\theta = \{m, b, \ell, k\}$ with $\theta \sim \mathcal{N}(\hat{\theta}, \sigma^2 I_p)$.

Barycenter computation. Barycenters of $M = 1000$ random parameter samples are obtained with a fixed-point method in Álvarez Esteban et al. [80], until the relative Frobenius norm $F^{\text{rel}} = \|\Sigma^{(k+1)} - \Sigma^{(k)}\|_F / \|\Sigma^{(k)}\|_F$ converges ($F^{\text{rel}} \le 1e^{-10}$).

For a condition x, write Σ_i, $i = 1, \dots, M$, for the covariances of the M jittered pushforward Gaussians at that condition. The Fréchet mean (μ_F, Σ_F) is computed as follows.

The mean component is closed form,

$$\mu_F = \frac{1}{M} \sum_{i=1}^{M} \mu_i. \tag{166}$$

The covariance is the Álvarez-Esteban fixed point, initialized at the *arithmetic* mean, and iterated as

$$\Sigma_F^{(0)} = \frac{1}{M} \sum_{i=1}^{M} \Sigma_i, \tag{167}$$

$$\bar{T}^{(k)} = \frac{1}{M} \sum_{i=1}^{M} \left(\Sigma_F^{(k)}\right)^{-1/2} \left[\left(\Sigma_F^{(k)}\right)^{1/2} \Sigma_i \left(\Sigma_F^{(k)}\right)^{1/2}\right]^{1/2} \left(\Sigma_F^{(k)}\right)^{-1/2}, \tag{168}$$

$$\Sigma_F^{(k+1)} = \bar{T}^{(k)} \, \Sigma_F^{(k)} \, \bar{T}^{(k)}. \tag{169}$$

Every matrix square root is evaluated spectrally on the symmetrized argument, with eigenvalues clamped before the root is taken:

$$S^{1/2} = U \operatorname{diag}\left(\sqrt{\max(\lambda_j, 0)}\right) U^\top, \qquad S^{-1/2} = U \operatorname{diag}\left(\max(\lambda_j, 10^{-10})^{-1/2}\right) U^\top, \tag{170}$$

where $U \operatorname{diag}(\lambda) U^\top$ is the eigendecomposition of $\frac{1}{2}(S + S^\top)$.

The iteration stops at the first k with

$$\frac{\left\|\Sigma_F^{(k+1)} - \Sigma_F^{(k)}\right\|_F}{\left\|\Sigma_F^{(k)}\right\|_F} < \tau, \qquad \tau = 10^{-10}, \tag{171}$$

The fixed-point iteration converged within 40 iterations across all settings. The norms are Frobenius norms taken over the *stacked* tensor of all conditions, so the criterion is a single pooled residual rather than one test per condition. The accepted iterate is symmetrized and factorized with a small jitter,

$$\Sigma_F \leftarrow \tfrac{1}{2}\left(\Sigma_F + \Sigma_F^\top\right), \qquad L_F L_F^\top = \Sigma_F + 10^{-14} I. \tag{172}$$

For each dimension d, we calculate $\mathrm{Tr}(G_{\hat{\theta}}(x))$ and $V_F^{\sigma^2}(x)$ per condition.

We calculate Fréchet variance from a Monte-Carlo sampling over $M = 1000$ draws of the last-layer perturbation. A single set of M draws is shared across all conditions and across $\sigma \in \{3{\times}10^{-4}, 10^{-3}, 3{\times}10^{-3}, 10^{-2}\}$ (common random numbers), and is redrawn independently for each of the three seeds of σ. For each σ and x, we independently fit the ordinary least squares regression

$$V_F^{\sigma^2}(x)/\sigma^2 = a(x) + b(x)\,\sigma^2 \tag{173}$$

and report $\mathrm{Tr}(G_{\hat{\theta}}(x))/\hat{a}(x) - 1$.

We find that the leading-order of the bound grows from 1 to 1.7 fold of the Fréchet variance as dimension d grows from 1 to 50 (Figure S1).

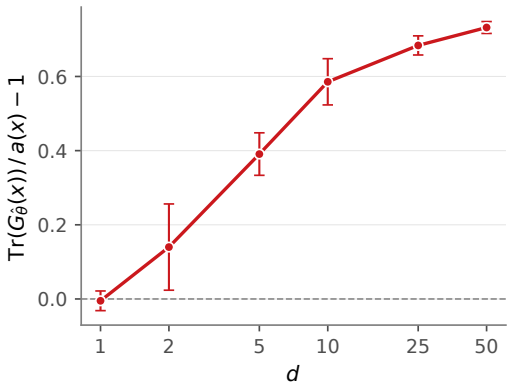

Figure S1: Per-dimension gap in the leading order. Bars are one standard deviation over conditions (x).

E.2 Multi-dimensional Gaussian affine transport

To test the impact of bound looseness in higher dimensions, we examined active learning performance in the affine transport of higher-dimensional Gaussians in $d \in \{5, 10, 25, 50\}$.

Mean field. For $k = 0, \ldots, d-1$,

$$\mu_k(x) = \alpha\, e^{-\beta x_1} \cos\Big(\frac{2\pi k}{d} + \pi x_2\Big), \qquad \alpha = 2,\ \beta = 6.$$

All d coordinates are driven by the same two scalars, so the mean field stays intrinsically two-dimensional at every d.

Covariance field. The covariance is a smooth, full-rank random field built from $r = d$ random Fourier features:

$$\Sigma(x) = s_d^2\big(A(x)A(x)^\top + \varepsilon I_d\big), \qquad A(x) = I_d + \sum_{j=1}^{d} f_j(x)\, B_j,$$

$$f_j(x) = \sqrt{2/d}\, \cos\big(\omega_j^\top x + \phi_j\big),$$

with $\varepsilon = 10^{-3}$ and

$$(B_j)_{ab} \overset{\text{iid}}{\sim} \mathcal{N}(0, 1/d), \qquad \omega_j \sim \mathcal{N}\big(0, (2\pi\kappa)^2 I_2\big), \qquad \phi_j \sim \mathcal{U}[0, 2\pi), \qquad \kappa = 1.$$

The parameters $\{B_j, \omega_j, \phi_j\}_{j=1}^d$ are drawn once per d, in that order, from a dedicated random generator seeded with d. They are independent of the experiment seed, so every seed sees the same ground-truth field. The scale s_d is calibrated so that the average per-coordinate variance is $\sigma_0^2 = 1$. Concretely, we draw $M = 2000$ calibration conditions $x^{(m)} \sim \mathcal{U}[0,1]^2$ from a generator with a fixed seed and set

$$s_d^2 = \sigma_0^2 \Big/ \Big(\frac{1}{Md} \sum_{m=1}^{M} \mathrm{tr}\big(A(x^{(m)})A(x^{(m)})^\top + \varepsilon I_d\big)\Big).$$

With this normalization, the mean and covariance contribute comparably to the variability of the data at every d: the covariance accounts for 41%, 55%, 54% and 51% of the pool's Bures–Wasserstein variability at $d = 5, 10, 25, 50$. The condition number of $\Sigma(x)$ grows with d, reaching about 3×10^3 at $d = 50$.

Pool, test set and initial set. Let $n_{\text{pool}} = \max(200, 40d)$, $n_{\text{test}} = \max(100, 40d)$ and $m = \mathrm{round}(\sqrt{n_{\text{pool}}})$.

- *Pool.* Take the one-dimensional grid $t_i = i/(m-1)$, $i = 0, \ldots, m-1$, and perturb each node independently by $\mathcal{U}(-0.05/m,\ 0.05/m)$. The pool is the Cartesian product of this perturbed grid with itself (m^2 conditions). If m^2 exceeds n_{pool}, n_{pool} of them are kept uniformly at random. The perturbation and the subsample depend on the experiment seed.

- *Test set.* A fixed, unperturbed $m_t \times m_t$ grid with $m_t = \mathrm{round}(\sqrt{n_{\text{test}}})$ and nodes equally spaced on $[\frac{1}{2m}, 1 - \frac{1}{2m}]$, i.e. offset from the pool by half a cell. It is the same for every seed.
- *Initial labelled set.* $b_0 = 3d$ pool conditions drawn uniformly at random without replacement, depending on the experiment seed.

The resulting sizes are:

d	5	10	25	50
pool	196	400	1000	2000
test	196	400	1024	2025
initial b_0	15	30	75	150

Model, training and selection. The model has the same two-network structure as in 2D: a mean network with d outputs and a Cholesky network with $d(d+1)/2$ outputs (softplus on the d diagonal entries). Both are ReLU MLPs with 2 hidden layers of width 64 at $d = 5$ and 3 hidden layers of width 256 for $d \geq 10$. In each round, the networks are re-initialized and trained full-batch on the labeled conditions with the Bures–Wasserstein loss $\frac{1}{|S|}\sum_{x\in S} W_2^2\big(\mathcal{N}(\mu(x), \Sigma(x)), \mathcal{N}(\hat{\mu}(x), \hat{\Sigma}(x))\big)$. Training uses Adam (learning rate 10^{-3}, weight decay 10^{-2}, gradient-norm clipping at 1) for at most 2000 steps, with early stopping allowed after step 1000 once the training loss has not improved for 10 consecutive steps. Each d is run with 50 seeds, which set the pool perturbation and subsample, the initial set, the network initializations and the random baseline. Within a seed, all methods share the pool, test set, and initial set.

Results With $B_0 = 3d, B = 4$, we see the advantage of tNTK over NTK up to $d = 50$ (Table S6).

Table S6: d-dimensional Gaussian benchmark with a full-rank covariance field: test W_2^2 (total) after the final active-learning round, starting from $b_0 = 3d$ labeled conditions and acquiring 5 rounds of $b = 4$; mean ± standard error over 50 seeds; lower is better, best per column in bold.

Method	$d = 5$	$d = 10$	$d = 25$	$d = 50$
tNTK	$\mathbf{0.1164} \pm 0.0014$	$\mathbf{0.7457} \pm 0.0163$	$\mathbf{1.1017} \pm 0.0169$	$\mathbf{1.3109} \pm 0.0123$
NTK	0.1469 ± 0.0028	0.7942 ± 0.0141	1.1346 ± 0.0157	1.4449 ± 0.0141
Random	0.2082 ± 0.0131	0.7763 ± 0.0287	1.1150 ± 0.0222	1.4214 ± 0.0163

E.3 Results on tNTK computation

In this section, we report compute time and the Monte Carlo stability of the tNTK.

E.3.1 Kernel compute time

We report the compute time for the tNTK calculation for the conditional Monge gap and flow matching models reported in the main text in Table S7.

E.3.2 Kernel stability to the number of Monte Carlo samples

We tested the Monte Carlo stability of tNTK with varying numbers of samples. We measured the stability metric of the relative Frobenius error ($\|K^{\text{full}} - K^{\text{downsampled}}\|_F / \|K^{\text{full}}\|_F$) and overlap in the selected samples over 20 independent draws and report the mean in Table S8. Specifically, among the reported runs in the main paper, the MC stability is measured from the fold 0, seed 0, AL iteration 1 model. For conditional Monge gap, the reference uses all control cells (K562 3,359, BMDM 25,952), which is also what the reported runs use. Draws are without replacement, so $N \geq 3{,}359$ is not available for K562 (–). For conditional flow matching, the source samples are fresh $\mathcal{N}(0, I)$ draws, the reference uses $N = 65{,}536$, and the reported runs use $N = 4{,}096$.

Table S7: Wall-clock cost of the tNTK kernel in the reported active-learning runs. Per AL iteration ≥ 1: kernel construction time (median, with 10th–90th percentile), selection time and model-fit time (medians, seconds), and the kernel's share of the iteration, $t_{\text{kernel}}/(t_{\text{kernel}} + t_{\text{select}} + t_{\text{fit}})$ (median, %). N is the number of source samples the kernel averages over (conditional Monge: all control cells; CellFlow: fresh $\mathcal{N}(0, I)$ draws). n is the number of AL iterations measured (fit and share use only iterations that trained the model, excluding ones restored from a checkpoint). Times come from the production runs' own instrumentation on shared GPU nodes, including JIT compilation on the first iteration.

Dataset	$B_0 + B$	N	Kernel (s)	Select (s)	Fit (s)	Kernel (%)	n
Conditional Monge							
K562-Chemical	10 + 1	3,359	16.4 [11.6, 31.3]	0.2	1686	1.0	247
	10 + 5	3,359	21.1 [12.5, 29.9]	1.5	2380	0.8	177
	20 + 5	3,359	15.7 [12.5, 28.2]	1.3	2747	0.6	153
	20 + 10	3,359	37.6 [34.2, 45.1]	0.6	4320	0.9	250
	30 + 5	3,359	45.7 [36.8, 51.8]	0.3	4151	1.1	250
	30 + 10	3,359	44.0 [35.4, 51.6]	0.7	5192	0.7	250
BMDM-Genetic	10 + 1	25,952	167.1 [120.3, 199.1]	1.1	1678	8.5	250
	10 + 5	25,952	160.8 [117.3, 222.1]	6.3	3548	4.3	250
	20 + 5	25,952	129.1 [97.5, 179.2]	6.1	4160	3.0	250
	20 + 10	25,952	193.5 [150.7, 232.9]	0.6	4699	3.8	250
	30 + 5	25,952	213.7 [186.7, 251.9]	0.3	4593	4.4	250
	30 + 10	25,952	253.7 [146.2, 361.3]	0.8	6209	4.0	272
CellFlow							
Radial flow	5 + 1	4,096	30.9 [28.3, 39.6]	1.3	299	9.5	150
	5 + 5	4,096	31.5 [28.4, 42.1]	7.7	380	7.6	120
	10 + 5	4,096	31.8 [27.8, 45.0]	6.2	359	8.2	150
Van der Pol oscillator	5 + 1	4,096	31.6 [26.2, 44.8]	1.2	229	13.1	150
	5 + 5	4,096	30.7 [25.5, 42.9]	6.4	278	10.1	150
	10 + 5	4,096	27.7 [24.8, 33.8]	6.6	255	10.1	150
CLINC 150	5 + 1	4,096	40.9 [34.6, 49.1]	1.5	922	4.1	150
	5 + 5	4,096	36.8 [30.1, 51.5]	7.6	1338	2.7	150
	10 + 5	4,096	39.2 [29.1, 59.6]	6.2	1725	2.2	196

Table S8: Monte Carlo stability of the tNTK kernel. For each N, the kernel is rebuilt from N source samples and compared with a reference kernel. *Err.* is the relative Frobenius error (%). *Selection %* is the percentage of conditions in the selected batch of size B that also appear in the batch selected with the reference kernel (out of B). Mean over 20 independent draws.

		N = 256		N = 1,024		N = 4,096		N = 16,384	
Dataset	$B_0 + B$	Err.	Selection %	Err.	Selection %	Err.	Selection %	Err.	Selection %
Conditional Monge									
K562-Chemical	10 + 1	2.52	100.0	1.16	100.0	–	–	–	–
	10 + 5	2.52	70.0	1.16	82.0	–	–	–	–
	20 + 5	4.04	100.0	1.56	100.0	–	–	–	–
	20 + 10	4.04	99.5	1.56	99.5	–	–	–	–
	30 + 5	4.84	97.0	2.01	100.0	–	–	–	–
	30 + 10	4.84	96.0	2.01	100.0	–	–	–	–
BMDM-Genetic	10 + 1	2.54	50.0	1.25	75.0	0.44	100.0	0.17	100.0
	10 + 5	2.54	33.0	1.25	73.0	0.44	97.0	0.17	100.0
	20 + 5	3.18	90.0	1.88	98.0	0.72	100.0	0.27	100.0
	20 + 10	3.18	86.5	1.88	91.5	0.72	100.0	0.27	100.0
	30 + 5	3.38	86.0	1.80	99.0	0.94	100.0	0.29	100.0
	30 + 10	3.38	88.0	1.80	94.5	0.94	99.5	0.29	100.0
CellFlow									
Radial flow	5 + 1	2.53	30.0	1.90	20.0	1.06	45.0	0.71	30.0
	5 + 5	2.68	78.0	2.02	83.0	1.24	80.0	1.03	84.0
	10 + 5	3.68	85.0	2.52	85.0	1.55	82.0	0.87	80.0
Van der Pol oscillator	5 + 1	3.62	100.0	2.72	100.0	1.47	100.0	0.99	100.0
	5 + 5	3.62	87.0	2.71	92.0	1.47	98.0	0.99	97.0
	10 + 5	3.50	92.0	2.44	89.0	1.44	89.0	0.87	88.0
CLINC 150	5 + 1	1.82	100.0	1.13	100.0	0.55	100.0	0.27	100.0
	5 + 5	1.85	74.0	1.16	87.0	0.56	92.0	0.27	98.0
	10 + 5	1.58	71.0	0.90	91.0	0.41	100.0	0.26	100.0

E.4 Additional baselines

We present additional baselines for the results presented in the main text. For affine transport of 2D Gaussian distributions, we also benchmarked against BatchBALD (NTK with greedy maximum determinant selection) and BADGE-style (NTK with k-means++ selection) adapted for the regression as in Holzmüller et al. [25] (Table S9). tNTK, with minimization of total posterior variance selection, outperforms all NTK-based baselines.

Table S9: 2D Gaussian: NTK-based batch-selection baselines. Decrease of W_2^2 relative to the initial training set, averaged over AL iterations 1–5 (units 10^{-3}; more negative is better), mean ± s.e.m. over 30 seeds.

Method	ΔW_2^2 (↓)
tNTK	$\mathbf{-431.58} \pm 26.68$
NTK (BAIT; NTK + total posterior variance minimization)	-364.15 ± 26.59
BatchBALD (NTK + maximum determinant)	-360.94 ± 26.30
BADGE-style (NTK + k-means++)	-405.88 ± 27.09
Random	-363.93 ± 26.09

We also benchmark tNTK in a larger-batch regime with LCMD selection, which Holzmüller et al. [25] has shown to perform the best in a large-batch scenario of $B_0 = 5$, $B = 20$ with 1 active learning iteration. In this scenario, the LCMD selection outperformed total posterior variance minimization, while tNTK outperformed NTK (Table S10). The result shows the importance of choosing the right selection algorithm in various active learning settings, as has been shown before [81]. We defer the investigation of the synergy between the tNTK and selection algorithms to future studies.

Table S10: 2D Gaussian: LCMD versus minimum total posterior variance (BAIT) as the batch-selection rule. Mean ± standard error over 50 seeds is shown. The two rules are matched runs (same seeds, initial set, and round-0 model). Random ignores the kernel and the selection rule, so its row is shared.

Kernel	LCMD	Min. total posterior variance
tNTK	$\mathbf{0.0573} \pm 0.0021$	0.0788 ± 0.0026
NTK	0.0643 ± 0.0025	0.2153 ± 0.0072
Random	–	0.1234 ± 0.0061

The experimental settings are identical to those presented in the main text unless otherwise mentioned, and are described in detail in Appendix D.3.

F Supplementary Figures

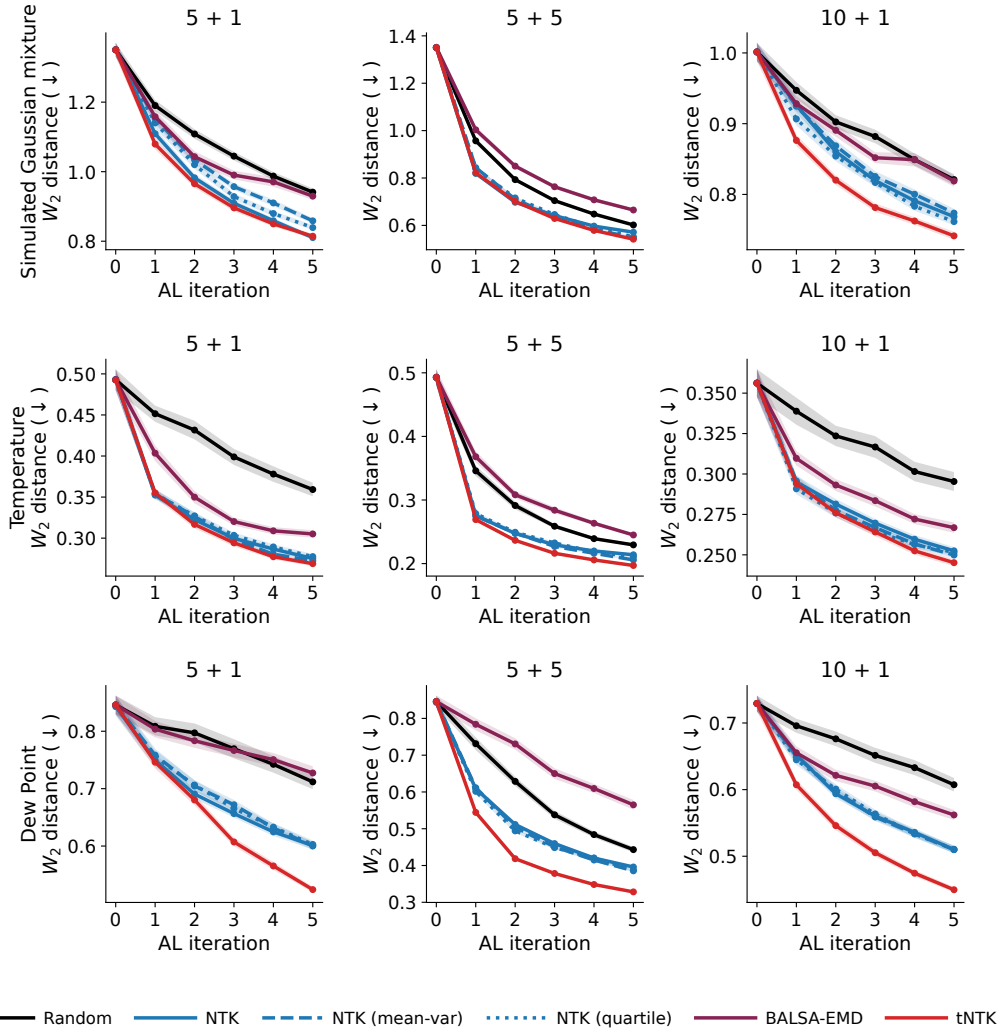

Figure S2: Wasserstein-2 distance over active learning iterations for the conditional quantile regression experiment. Rows correspond to datasets (Simulated Gaussian mixture, Temperature, Dew Point); columns correspond to budget configurations $B_0 + B$ (initial training set size + per-round batch size). Lines show mean across 50 runs (5 folds × 10 seeds), with shaded bands showing ± standard error. Random in black, NTK in blue, tNTK in red.

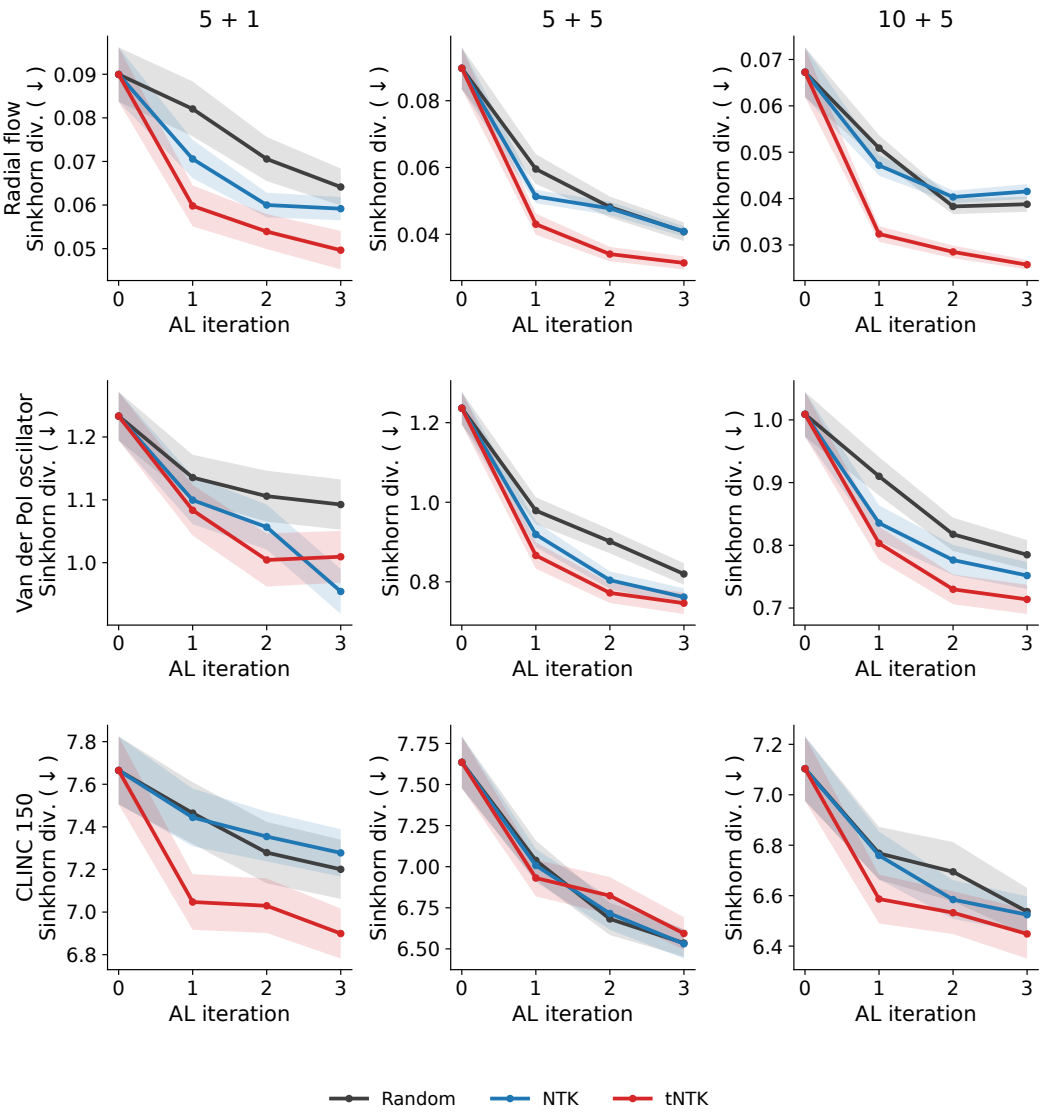

Figure S3: Sinkhorn divergence over active learning iterations for conditional flow matching. Rows correspond to datasets (radial flow, Van der Pol oscillator, CLINC150); columns correspond to budget configurations $B_0 + B$ (initial training set size + per-round batch size). Lines show mean across 50 runs (5 folds $\times$ 10 seeds), with shaded bands showing $\pm$ standard error. Random in black, NTK in blue, tNTK in red.

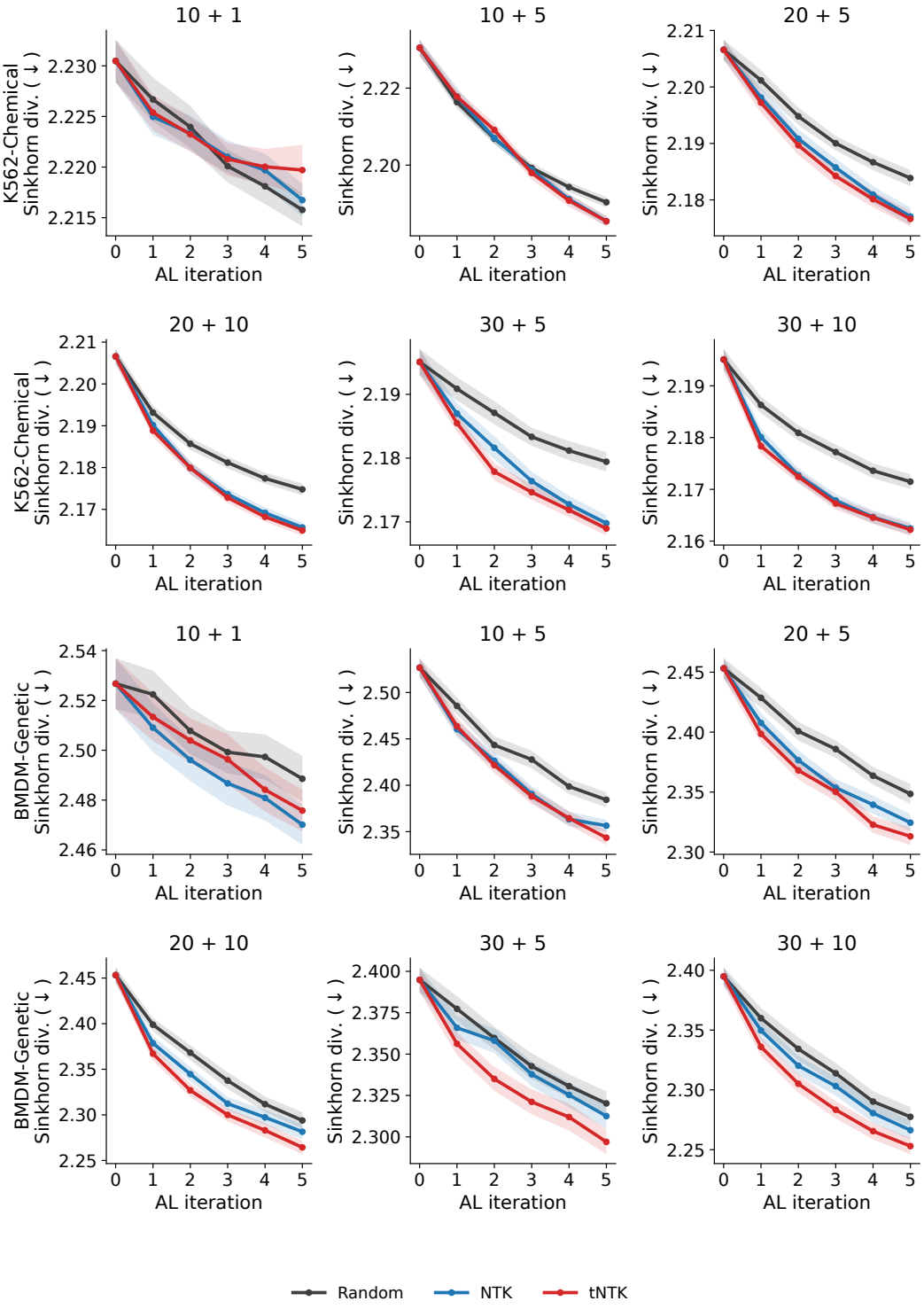

Figure S4: Sinkhorn divergence over active learning iterations for the conditional Monge gap. Rows correspond to datasets (K562-Chemical, BMDM-Genetic); columns correspond to budget configurations $B_0 + B$ (initial training set size + per-round batch size). Lines show mean across 50 runs (5 folds × 10 seeds), with shaded bands showing ± standard error. Random in black, NTK in blue, tNTK in red.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our abstract and introduction clearly discusses contributions, assumptions, and limitations made by this paper.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitation of our work in Discussion section that the application of tNTK in higher dimensions and other batch selection methods are not addressed by this paper.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: For each theoretical result, we provide the full assumptions, and also provide proof in Appendices that are referenced in the main text.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We attach anonymized source code to reproduce the main results, and provide sufficient details needed for reproduction in the manuscript.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide anonymized source code as supplementary material with instructions to reproduce all experiments. All datasets are publicly available with citations in Appendix D.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: All the relevant experimental details are provided in Appendix D.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We provide standard error in all tables and supplementary figures and describe it in the table captions and figure legends.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Compute resources required is provided in Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Individual items in the code of ethics are properly addressed or not applicable to this study, and are answered in more details in the rest of the checklist.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: Whereas this paper aims for the better generalization of generative models to unseen conditions, practical challenges remain in generalizing the framework to models with > 100 latent dimensions capable of generation of realistic images or text.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This paper poses no risk on release of data or models that has high risk of misuse.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited, and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All code and datasets used are properly cited with the license in Appendix D.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: This paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research did not involve LLMs.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.