Learning from unlabeled data for interacting particle system

Fei Lu
feilu@math.jhu.edu
Last updated: 2026-1-27

Abstract

Learning the dynamics of complex high-dimensional interacting particle systems is a fundamental task across various scientific disciplines. However, a common challenge in these applications is the unlabeled ensemble data collected at discrete time points, which lack trajectory information due to limitations in data-collection methods or privacy concerns. We overcome the challenge by using weak-form PDEs to construct trajectory-free loss functions. We employ an automatic reproducing kernel derived from the data and model to estimate the interaction potential, ensuring an RKHS tailored to the inverse problem.

1 Learning a single system from ensemble data

Consider the estimation of the interaction potential Φ and kinetic potential V in the interacting particle system of N particles with $W_t^i \in \mathbb{R}^d$ a standard Brownian motion:

$$dX_t^i = -\frac{1}{N}\sum_{i\neq j=1}^{N} \nabla\Phi(X_t^i - X_t^j)dt + \nabla V(X_t^i)dt + \sigma dW_t^i, \quad i = 1, \ldots N\,, \quad t \in [0, T] \tag{1}$$

from *unlabeled data* consisting of a sequence of sample ensembles denoted by

$$\mathcal{D} = \{\mathbf{X}_{t_\ell}^{1:M_\ell}\}_{\ell=1}^{L} \text{ with } \mathbf{X}_t = (X_t^1, \ldots, X_t^N), \qquad M_\ell \geqslant M, \quad \ell = 1, \ldots, L. \tag{2}$$

Here $\{t_\ell\}_{\ell=1}^L \subset [0, T]$ are time points, and M_ℓ is the number of samples at t_ℓ and may vary in ℓ. In particular, the data misses trajectory information in the sense that the positions $X_{t_\ell}^{i,m}$ and $X_{t_{\ell+1}}^{i,m}$ are not paired in data either since the label is unknown or they may come from different trajectories. Such types of data arise in various applications, such as opinion polls [?, ?, ?, ?], gene methylation [?, ?, ?, ?], and neuroscience [?, ?, ?], to name just a few.

The major challenge comes from the **missing trajectory information**. It prevents the applications of methods that use the derivatives dX_t^i/dt or their approximation, such as the widely used methods based on minimizing the mean-square prediction error, the likelihood of the trajectories, or the posterior in Bayesian methods [?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?]. In particular, the number N is often too small to use the mean-field equation for the inference (or the Fokker-Planck equation for estimating V when $\Phi \equiv 0$) [?, ?, ?, ?]. We overcome the challenge by constructing a trajectory-free loss function using weak-form PDEs. Based on this, we infer the potential functions by a novel automatic kernel regression method, and build a systematic theory on identifiability and minimax rates.

2 Trajectory-free loss functions using weak-form PDEs

2.1 The weak-form PDE for the empirical distribution

We overcome the challenge by constructing trajectory-free loss functions based on the weak-form PDE that characterizes the evolution of the empirical distribution of particles. The empirical distribution does not require the trajectory information of any particle, but it accounts for the temporal evolution of the ensemble.

Specifically, let $\mu_t^N(x) := \frac{1}{N}\sum_{i=1}^N \delta_{X_t^i - x}$ be the empirical distribution of the particles at time t, where δ_x is the Dirac delta function at $x \in \mathbb{R}^d$. It evolves according to the following weak-form PDE:

$$\partial_t \mu_t^N = -\nabla \cdot \left[\mu_t^N \nabla\big(\Phi * \mu_t^N + V\big)\right] + \frac{\sigma^2}{2}\Delta \mu_t^N + \sigma m^{\mathbf{X}_t}, \quad x \in \mathbb{R}^d, t \in [0,T], \tag{3}$$

where $m_t^{\mathbf{X}}$ is a $\mathbf{X}_t$-dependent measure that leads to a martingale term. Equivalently, for any test function $f \in C_b^2(\mathbb{R}^d)$,

$$d\frac{1}{N}\sum_i f(X_t^i) = \frac{1}{N}\sum_i \big[\nabla f(X_t^i)\big(\nabla V(X_t^i) + \frac{1}{N}\sum_j \nabla\Phi(X_t^i - X_t^j)\big) + \frac{\sigma^2}{2}\Delta f(X_t^i)\big]dt + \sigma m^{\mathbf{X}_t}(f)$$

with

$$m^{\mathbf{X}_t}(f) := \frac{1}{N}\sum_i \nabla f(X_t^i) dW_t^i.$$

In other words, the weak-form PDE is derived from the ODE/SDE of the particle system (1) by applying the chain rule or the Itô formula to $\langle \mu_t^N, f\rangle$.

Importantly, it is not the Liouville or Fokker-Planck equation of the ODE or SDE, which characterizes the evolution of the joint probability density of the particles on $\mathbb{R}^{Nd}$. Instead, it is a PDE for the evolution of the empirical distribution of the particles on $\mathbb{R}^d$. In particular, when the particle system is stochastic, it is a non-closed stochastic PDE with a state-dependent martingale term $m^{\mathbf{X}_t}(f)$.

The weak-form PDE characterizes the evolution of the empirical distribution of the particles. It does not require the trajectory information of any single particle. It takes into account the full information of the data, which consists of a sequence of ensembles. It provides the foundation for our construction of trajectory-free loss functions.

2.2 Trajectory-free loss functions.

We propose a new trajectory-free loss function for learning the potential functions from the unlabeled ensemble data (2). For simplicity, we assume that $M_\ell \equiv M$ for all ℓ. Our trajectory-free loss function has the following form:

$$\begin{aligned}\mathcal{E}_{\mathcal{D}}(\Phi, V) =& \frac{1}{M}\sum_{m,\ell=1}^{M,L} \mathcal{E}_{\mathbf{X}_{t_\ell}^m, \mathbf{X}_{t_{\ell+1}}^m}(\Phi, V), \qquad \text{with} \\ \mathcal{E}_{\mathbf{X}_{t_\ell}, \mathbf{X}_{t_{\ell+1}}}(\Phi, V) =& \underbrace{\int |\nabla V + \nabla\Phi * \mu_{t_\ell}^N|^2 \mu_{t_\ell}^N dx \Delta t}_{Dissipation} + \underbrace{\sigma \int [\Delta V + \Delta\Phi * \mu_{t_\ell}^N]\mu_{t_\ell}^N dx \Delta t}_{Diffusion} \\ & \underbrace{- 2\int [V + \Phi * \mu_t^N]\mu_t^N dx\Big|_{t_\ell}^{t_{\ell+1}}}_{Energy\ change},\end{aligned} \tag{4}$$

where by the definition of μ_t^N, the three integrals in the dissipation, diffusion, and energy change terms can be computed as:

$$J_{diss} = \int |\nabla V + \nabla \Phi * \mu_t^N|^2 \mu_t^N dx = \frac{1}{N}\sum_{i=1}^N \big|\nabla V(X_t^i) + \frac{1}{N}\sum_{j=1}^N \nabla \Phi(X_t^i - X_t^j)\big|^2,$$

$$J_{diff} = \int |\Delta V + \Delta \Phi * \mu_t^N| \mu_t^N dx = \frac{1}{N^2}\sum_{i,j=1}^N \big[\Delta V(X_t^i) + \Delta \Phi(X_t^i - X_t^j)\big],$$

$$J_{energy} = \int [V + \Phi * \mu_t^N] \mu_t^N dx = \frac{1}{N^2}\sum_{i,j=1}^N \big[V(X_t^i) + \Phi(X_t^i - X_t^j)\big].$$

.

It is **trajectory-free** because the first two terms in $\mathcal{E}_{\mathbf{X}_{t_\ell : t_{\ell+1}}}(\Psi, V)$ don't use any trajectory information, and the last term only requires the sequential information of the empirical measures $\mu_{t_\ell}^N \to \mu_{t_{\ell+1}}^N$, but not the trajectory of any single particle.

We derive this loss function using $\Phi * \mu_t^N + V$ as the test function in the weak-form PDE in (3). It shares the idea of using weak solutions in [?, ?, ?] to avoid the need for a strong solution, which does not exist in our setting. This test function solves the difficult problem of selecting a large class of test functions, particularly when the state X_t^i is high-dimensional. It is tailored for inferring function parameters in Wasserstein gradient flows.

3 Computation: neural network regression

3.1 Neural network regression using the self-test loss

We employ neural networks to approximate the unknown potentials V and Φ. Let $V_\theta : \mathbb{R}^d \to \mathbb{R}$ and $\Phi_\eta : \mathbb{R}^d \to \mathbb{R}$ be parametrized by deep neural networks with weights θ and η. To ensure the physical consistency of the interaction potential, we enforce symmetry by setting $\Phi_\eta(x) = \frac{1}{2}(\tilde{\Phi}_\eta(x) + \tilde{\Phi}_\eta(-x))$, where $\tilde{\Phi}_\eta$ is the raw network output.

Unlike strong-form methods that require estimating time derivatives (velocities) $\frac{dX_t}{dt}$—which is impossible with unlabeled ensemble data—our approach utilizes the time-integrated weak form. The loss function $\mathcal{E}_{\mathcal{D}}(\Phi_\eta, V_\theta)$ is computed using the discrete ensembles directly.

The spatial gradients $\nabla V_\theta(x)$ and $\nabla \Phi_\eta(x)$ required in the loss function are computed via Automatic Differentiation (AD) with respect to the inputs x. This avoids numerical errors associated with spatial finite differences on scattered data.

Algorithm 1: Neural Network Estimation via Self-Test Loss

1. **Input:** Unlabeled ensembles $\mathcal{D} = \{\mathbf{X}_{t_\ell}^{1:M}\}_{\ell=1}^L$.

2. **Initialization:** Initialize network parameters θ, η.

3. **Optimization Loop:**

   - Sample a batch of time indices $\mathcal{I}_{batch} \subset \{1, \ldots, L-1\}$.
   - For each $\ell \in \mathcal{I}_{batch}$:
     - **Dissipation Term:** Compute the drift vector for all particles in the ensemble at t_ℓ:

$$D(X_{t_\ell}^i) = \nabla V_\theta(X_{t_\ell}^i) + \frac{1}{N}\sum_{j \neq i} \nabla \Phi_\eta(X_{t_\ell}^i - X_{t_\ell}^j)$$

    Compute the dissipation integral estimate (Monte Carlo):

$$J_{diss} = \frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{i=1}^{N}\|D(X_{t_\ell}^{i,m})\|^2\Delta t_\ell$$

  - **Diffusion Term:** Compute the Laplacians for all particles at t_ℓ:

$$L(X_{t_\ell}^i) = \Delta V_\theta(X_{t_\ell}^i) + \frac{1}{N}\sum_{j\neq i}\Delta\Phi_\eta(X_{t_\ell}^i - X_{t_\ell}^j)$$

    Compute the diffusion integral estimate (Monte Carlo):

$$J_{diff} = \frac{\sigma}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{i=1}^{N}L(X_{t_\ell}^{i,m})\Delta t_\ell$$

  - **Energy Term:** Compute the energy change between snapshots t_ℓ and $t_{\ell+1}$:

$$E(t) = \frac{1}{N}\sum_{i=1}^{N}V_\theta(X_t^i) + \frac{1}{2N^2}\sum_{i,j}\Phi_\eta(X_t^i - X_t^j)$$

    Calculate difference: $\Delta J_{energy} = \frac{1}{M}\sum_{m=1}^{M}[E(t_{\ell+1}) - E(t_\ell)]$.

- **Loss Calculation:** Construct the empirical loss:

$$\widehat{\mathcal{L}}(\theta,\eta) = \sum_{\ell\in\mathcal{I}_{batch}}\left(J_{diss}^{(\ell)} + J_{diff}^{(\ell)} + 2\Delta J_{energy}^{(\ell)}\right)$$

- **Backpropagation:** Compute gradients $\nabla_{\theta,\eta}\widehat{\mathcal{L}}$ and update weights using Adam/NAdam.

4. **Output:** Learned potentials $V_{\hat{\theta}}, \Phi_{\hat{\eta}}$.

Crucially, this algorithm only pairs distributions $\mu_{t_\ell}^N$ and $\mu_{t_{\ell+1}}^N$ via the energy difference term; it never assumes $X_{t_\ell}^i$ and $X_{t_{\ell+1}}^i$ correspond to the same particle.

3.2 Numerical examination of accuracy

After obtaining the estimated potentials $V_{\hat{\theta}}$ and $\Phi_{\hat{\eta}}$ from the optimization process, we numerically assess their accuracy. In a synthetic data setting where the true potentials V_{true} and Φ_{true} are known, we perform a quantitative comparison. The accuracy is evaluated by computing the relative L^2 error over a set of test points $\{\mathbf{X}_{tl}^m\}_{l=1,m=1}^{L_{test},M_{test}}$ sampled from the same distribution as the training data. The errors for the potential functions and their gradients are calculated as:

$$\text{Error}(\Phi) = \frac{\sqrt{\sum_{l=1}^{L_{test}}\sum_{m=1}^{M_{test}}\sum_{i,j}|\Phi_{\hat{\eta}}(r_{lijm}) - \Phi_{true}(r_{lijm})|^2}}{\sqrt{\sum_{l=1}^{L_{test}}\sum_{m=1}^{M_{test}}\sum_{i,j}|\Phi_{true}(r_{lijm})|^2}}, r_{lijm} := X_{t_l}^{j,m} - X_{t_l}^{i,m}$$

$$\text{Error}(V) = \frac{\sqrt{\sum_{l=1}^{L_{test}}\sum_{m=1}^{M_{test}}\sum_{i}|V_{\hat{\theta}}(X_{t_l}^{i,m}) - V_{true}(X_{t_l}^{i,m})|^2}}{\sqrt{\sum_{l=1}^{L_{test}}\sum_{m=1}^{M_{test}}\sum_{i}|V_{true}(X_{t_l}^{i,m})|^2}}.$$

Additionally, we can visually compare the learned potentials and their gradients against the true functions by plotting them over a 1D or 2D grid. This provides a qualitative assessment of the learning performance.