A DIFFUSIVE CLASSIFICATION LOSS FOR LEARNING ENERGY-BASED GENERATIVE MODELS

Louis Grenioux[⋆,1,3] **& RuiKang OuYang**[⋆,2] **José Miguel Hernández-Lobato**[2]
[1]École polytechnique, [2]University of Cambridge, [3]Flatiron Institute

ABSTRACT

Score-based generative models have recently achieved remarkable success. While they are usually parameterized by the score, an alternative way is to use a series of time-dependent energy-based models (EBMs), where the score is obtained from the negative input-gradient of the energy. Crucially, EBMs can be leveraged not only for generation, but also for tasks such as compositional sampling or building Boltzmann Generators via Monte Carlo methods. However, training EBMs remains challenging. Direct maximum likelihood is computationally prohibitive due to the need for nested sampling, while score matching, though efficient, suffers from mode blindness. To address these issues, we introduce the *Diffusive Classification* (`DiffCLF`) objective, a simple method that avoids blindness while remaining computationally efficient. `DiffCLF` reframes EBM learning as a supervised classification problem across noise levels, and can be seamlessly combined with standard score-based objectives. We validate the effectiveness of `DiffCLF` by comparing the estimated energies against ground truth in analytical Gaussian mixture cases, and by applying the trained models to tasks such as model composition and Boltzmann Generator sampling. Our results show that `DiffCLF` enables EBMs with higher fidelity and broader applicability than existing approaches.

1 INTRODUCTION

Probabilistic modeling is a cornerstone of modern machine learning, providing a principled framework to capture complex data distributions and to generate realistic samples. A classical approach is density estimation, often carried out by explicitly modeling it using Energy-Based Models (EBMs), where the density is parametrized as the exponential of the negation of a learnable function, referred to as the energy (LeCun et al., 2006; Kim & Bengio, 2016; Nijkamp et al., 2019; Du & Mordatch, 2019; Grathwohl et al., 2020; Che et al., 2020; Song & Kingma, 2021). While conceptually appealing, EBMs are notoriously hard to train due to the intractable normalizing constant that prevents maximum likelihood estimation, forcing reliance on costly sampling procedures. Despite advances in amortized and efficient sampling (Du & Mordatch, 2019; Du et al., 2021; Grathwohl et al., 2021; Carbone et al., 2023; Senetaire et al., 2025), training EBMs remains computationally challenging.

A popular alternative avoids the difficulty of modeling energies directly by targeting their negative gradient, the score function. Score matching methods (Hyvärinen, 2005) leverage the fact that the intractable normalizing constant disappears upon differentiation, making the score easier to estimate. This approach became especially prominent with the advent of score-based generative models such as Diffusion Models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021b) and Stochastic Interpolants (Albergo & Vanden-Eijnden, 2023; Albergo et al., 2023), where a forward noising process gradually transforms data into pure noise (or another tractable distribution), and sampling is achieved by reversing this process through a denoising dynamic. Crucially, these denoising dynamics depend on the score of the perturbed data distribution, which can be efficiently learned using Denoising Score Matching (Sohl-Dickstein et al., 2015; Song et al., 2021b).

[⋆]Equal contribution. The order is randomly assigned and randomly reshuffled in different versions of the paper to reflect this equal contribution. Correspondence to <lgrenioux@flatironinstitute.org> and <ro352@cam.ac.uk>.

Beyond generating samples, estimating the underlying energy function, rather than only the score, enables a range of downstream applications. Examples include building Boltzmann Generators from generative models via Monte Carlo methods (Phillips et al., 2024; Zhang et al., 2025a) and composing multiple models (Du et al., 2023; Skreta et al., 2025b;a; Thornton et al., 2025; He et al., 2025b), both of which fundamentally require access to energies. While prior works (Skreta et al., 2025b;a; Zhang et al., 2025a; He et al., 2025b) focus on estimating marginal density ratios using only scores, they typically rely on assumptions such as perfectly learned scores or approximations of transition kernels in small time steps. In contrast, direct access to the energy often leads to improved performance.

While several works (Salimans & Ho, 2021; Phillips et al., 2024; Thornton et al., 2025) have explored training energy-based generative models through score-based objectives, yielding approximations of the energies, these methods face significant limitations. Chief among them is mode blindness, where the relative proportions of disjoint high-density regions are misrepresented (Wenliang & Kanagawa, 2021; Zhang et al., 2022; Shi et al., 2024). Recent efforts aim to recover energies of diffusion models directly, but they often demand heavy computation or fragile hyperparameter tuning (Gao et al., 2021; Zhang et al., 2023; Schröder et al., 2023; Zhu et al., 2024).

Our contributions. We address the problem of training energy-based generative models in a way that enables direct downstream use of energies. Our main contributions are:

- We introduce the *Diffusive Classification* (`DiffCLF`) objective, which reframes log-density estimation as a **supervised classification problem**. `DiffCLF` is lightweight, flexible, and can be seamlessly combined with classical score-matching objectives.
- We prove that `DiffCLF` consistently recovers the ground-truth distribution at optimality and, unlike score-based methods, is *not mode-blind*.
- We establish connections between `DiffCLF` and prior approaches that exploit temporal correlations in stochastic processes to learn energy-based models.
- We demonstrate the effectiveness of `DiffCLF` across different generative processes and for a range of downstream tasks, including building Boltzmann Generators and model composition.

2 PRELIMINARY

2.1 GENERAL FRAMEWORK

This work considers stochastic processes on $\mathbb{R}^d$ for $t \in [0, T]$ of the form

$$Y_t = X_t + \gamma(t)Z \ , \tag{1}$$

where $\gamma \in C^2([0, T])$, T denotes the terminal time which may be infinite, $(X_t)_t$ is a stochastic process from which samples can be drawn at any time t, and Z is a standard Gaussian noise independent of $(X_t)_t$. Let q_t denote the marginal density of X_t and p_t the marginal density of Y_t. This framework serves as a generic setting, with concrete instances and applications provided later. The objective is to estimate the densities $(p_t)_t$ up to a normalizing constant, given access only to samples from X_t. To achieve this, a parametric family of energy-based models is introduced

$$p_t^\theta(y_t) = \exp(-\mathrm{U}_t^\theta(y_t))/\mathcal{Z}_t^\theta, \quad \mathcal{Z}_t^\theta = \exp(\mathrm{F}_t^\theta) = \int \exp(-\mathrm{U}_t^\theta(y_t))\mathrm{d}y_t \ , \tag{2}$$

where $\mathrm{U}^\theta : [0, T] \times \mathbb{R}^d \to \mathbb{R}$ is the energy function, parameterized by $\theta \in \Theta$ (for instance, neural networks), and $\mathcal{Z}_t^\theta$ is the associated normalizing constant, which is intractable in general. A natural approach to estimating this model is *Maximum Likelihood* (ML) which writes as

$$\mathcal{L}_{\mathrm{ML}}(\theta) = \mathbb{E}_t\left[\mathcal{L}_{\mathrm{ML}}(\theta; t)\right], \quad \mathcal{L}_{\mathrm{ML}}(\theta; t) = -\mathbb{E}_{p_t}[\log p_t^\theta(Y_t)] = \mathbb{E}_{p_t}[\mathrm{U}_t^\theta(Y_t)] + \log \mathcal{Z}_t^\theta \ . \tag{3}$$

Given $t \in [0, T]$, taking gradients of $\mathcal{L}_{\mathrm{ML}}(\cdot; t)$ with respect to θ yields

$$\nabla_\theta \mathcal{L}^{\mathrm{ML}}(\theta; t) = \mathbb{E}_{p_t}[\nabla_\theta \mathrm{U}_t^\theta(Y_t)] - \mathbb{E}_{p_t^\theta}[\nabla_\theta \mathrm{U}_t^\theta(Y_t)] \ . \tag{4}$$

The difficulty of this approach lies in the second term, which requires sampling from p_t^θ. Since this distribution can be as complex as the original data distribution q_t, sampling typically demands expensive Monte Carlo methods, making ML estimation impractical.

An alternative is to exploit the structure of Equation (1) and apply *Denoising Score Matching* (DSM) (Sohl-Dickstein et al., 2015; Song et al., 2021b) which aims at learning the gradient of the log-density (the score) to recover the energy up to an additive constant as a by-product. Let $p_t(\cdot|x_t)$ be the density of Y_t conditional on $X_t = x_t$. By construction, for all $t \in [0, T]$ and $x_t, y_t \in \mathbb{R}^d$

$$p_t(y_t|x_t) = \mathcal{N}(y_t; x_t, \gamma^2(t)\mathrm{I}_d) \ . \tag{5}$$

Using that $p_t(y_t) = \int p_t(y_t|x_t)q_t(x_t)\mathrm{d}x_t$, the score [1] can then be expressed as

$$\nabla \log p_t(y_t) = \mathbb{E}\left[\nabla_{y_t} \log p_t(y_t|X_t) \mid Y_t = y_t\right] = \mathbb{E}\left[-\frac{y_t - X_t}{\gamma^2(t)}\middle| Y_t = y_t\right] \ . \tag{6}$$

Equation (6) is often referred to as the Tweedie's formula (Efron, 2011). This characterization shows that the score is a conditional expectation over the posterior distribution of X_t given Y_t. Building on this, DSM defines the following mean square objective, whose minimum is attained by Equation (6),

$$\mathcal{L}_{\mathrm{DSM}}(\theta) = \mathbb{E}_t\left[\mathcal{L}_{\mathrm{DSM}}(\theta; t)\right], \quad \mathcal{L}_{\mathrm{DSM}}(\theta; t) = \mathbb{E}\left[\left\|\nabla \log p_t^\theta(Y_t) - \nabla \log p_t(Y_t|X_t)\right\|^2\right], \tag{7}$$

where $X_t \sim q_t$ and $Y_t \sim p_t(\cdot|X_t)$. Note that various heuristics exist for designing the time distribution see for instance (Song et al., 2021b; Karras et al., 2022; Kingma & Gao, 2023). The DSM objective doesn't require to compute the normalizing constant or to sample from the model.

The framework introduced in Equation (1) underlies many widely used generative models. The core idea is to generate new samples via a Markov process, typically formulated as a *Stochastic Differential Equation* (SDE), whose marginals coincide with those of Y_t. Crucially, constructing such processes relies on access to the score function $\nabla \log p_t$. Below, two concrete and widely used instances of this framework are presented.

Example 1 : Diffusion Models In *Diffusion Models* (DMs) (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021b), one usually defines $(Y_t)_t$ through a noising (forward) SDE[2] $\mathrm{d}Y_t = f(t)Y_t\mathrm{d}t + g(t)\mathrm{d}W_t$ starting from $Y_0 \sim p_0$, where $(W_t)_t$ is a standard Brownian motion. Using the notations of (Karras et al., 2022, Appendix B), the solution of the noising SDE can be written as $Y_t = S(t)Y_0 + S(t)\sigma(t)Z$ with $Z \sim \mathcal{N}(0, I)$, where $S(t) = \exp(\int_0^t f(u)\mathrm{d}u)$ and $\sigma(t)^2 = \int_0^t \left(g(u)/S(u)\right)^2 \mathrm{d}u$. The noising SDE fits Equation (1) with $X_t = S(t)X_0$ where $X_0 \sim p_0$ and $\gamma(t) = S(t)\sigma(t)$. Under mild conditions on π (Anderson, 1982), one can derive the corresponding denoising (backward) SDE

$$\mathrm{d}Y_t = \left[f(t)Y_t - g^2(t)\nabla \log p_t(Y_t)\right]\mathrm{d}t + g(t)\mathrm{d}\tilde{W}_t, \quad \text{from } Y_T \sim p_T \ , \tag{8}$$

where $(\tilde{W}_t)_t$ is a standard Brownian motion. Estimating the score function $\nabla \log p_t$ via DSM (7) allows approximating this SDE which can be solved backward in time to produce samples from π.

Example 2 : Stochastic Interpolants *Stochastic Interpolants* (SIs) (Albergo & Vanden-Eijnden, 2023; Albergo et al., 2023) generalize diffusion models by constructing generative processes that interpolate between any two distributions, not necessarily with a Gaussian endpoint. In this setting, $X_t = I_t(X_0, X_1)$ with $X_0 \sim q_0$, $X_1 \sim q_1$, where datasets are available for both q_0 and q_1. The interpolation function I is chosen so that X_0 and X_1 are recovered at times $t = 0$ and $t = 1$, while $\gamma(0) = \gamma(1) = 0$ ensures that $Y_0 \sim q_0$ and $Y_1 \sim q_1$. Under mild assumptions, the dynamics of $(Y_t)_t$ are given by the SDE

$$\mathrm{d}Y_t = \left[v_t(Y_t) - \left(\dot{\gamma}(t)\gamma(t) + \frac{g(t)^2}{2}\right)\nabla \log p_t(Y_t)\right]\mathrm{d}t + g(t)\mathrm{d}W_t \ , \quad \text{from } Y_0 \sim p_0, \tag{9}$$

where $(W_t)_t$ is a standard Brownian motion, g is any positive function and v_t is defined via a conditional expectation as $v_t(y_t) = \mathbb{E}[\partial_t I_t(X_0, X_1) \mid Y_t = y_t]$. As in DSM, v_t and the score can be learned via simple regression objectives , and once approximated, the SDE can be integrated to generate new samples from q_1 [3] . Notably, DMs appear as a special case of this framework.

[1] For clarity, we write $\nabla \log p_t(y)$ (respectively $\nabla \log p_t(y|x)$) as shorthand for $\nabla_y \log p_t(y)$ (respectively $\nabla_y \log p_t(y|x)$) whenever no ambiguity arises.

[2] Popular choices for the function f and g include the *Variance Preserving* (VP) and *Variance Exploding* (VE), see Appendix B.2 or Song et al. (2021b) for details.

[3] The transport from q_1 to q_0 can be obtained by changing the sign of the diffusion coefficient in front of the score in SDE (9), i.e. $g(t)^2/2 \to -g(t)^2/2$, and integrating from $t = 1$ to $t = 0$ with $Y_1 \sim p_1$.

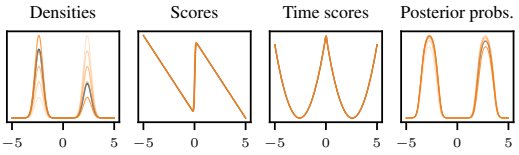


Figure 1: **Densities, scores, time-scores, and classification posterior probabilities.** Scores and time-scores are nearly identical across mixtures, while classification posteriors do vary.

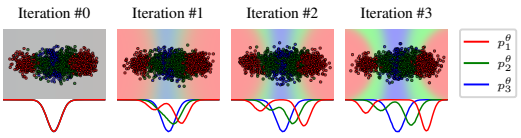


Figure 2: **Classification posteriors and EBMs during training.** Red, green, and blue dots are samples from $p_{t_1}, p_{t_2}, p_{t_3}$, with 1D marginals shown as curves. The background encodes classifier posterior probabilities (11).

2.2 Why modeling energies ?

As illustrated by DMs and SIs, the most natural quantity to model is often the score $\nabla \log p_t$. However, directly learning the energies themselves brings several important advantages .

Boltzmann Generators. Boltzmann Generators (BGs) (Noé et al., 2019) turn trained generative models into samplers for a target Boltzmann distribution π (known up to a constant) by reweighting or Markov corrections using learned density surrogates. Recent works in DMs replace classical importance sampling and MCMC with annealed and sequential methods such as AIS (Zhang et al., 2024; 2025a), SMC (Phillips et al., 2024), and RE (Zhang et al., 2025b), exploiting intermediate densities $(p_t^\theta)_t$ to bridge a base distribution and π.

Compositionality. Accurate energy estimation enables training-free composition of generative models. Recent work shows that diffusion models trained on different targets can be combined to represent mixtures or products, and efficiently sampled using advanced sampling methods such as AIS (Skreta et al., 2025a;b), annealed Langevin dynamics (Du et al., 2023; Lee et al., 2023; Zhu et al., 2024), SMC (Thornton et al., 2025; He et al., 2025b), and RE (He et al., 2025c).

Free energy difference estimation. Estimating free-energy differences between states is a central challenge in statistical physics (Lelièvre et al., 2010). Classical methods such as Classical methods such as *Thermodynamic Integration* (TI) (Kirkwood, 1935) and *Multistate Bennett Acceptance Ratio* (MBAR) (Shirts & Chodera, 2008) rely on annealed sequences of intermediate potentials. Recent works (Máté et al., 2025; He et al., 2025a) construct these paths via energy-based SIs, providing a data-driven alternative that can outperform hand-crafted designs.

2.3 On the limitation of score-matching methods

While score-based methods learn energies through gradients, they suffer from a fundamental limitation known as score-matching blindness (Wenliang & Kanagawa, 2021; Zhang et al., 2022; Shi et al., 2024). Divergences based solely on scores, e.g. Fisher divergence, are invalid for distributions with disjoint supports, as zero distance does not imply equality (Zhang et al., 2022, Theorem 2). Intuitively, scores capture only local, intra-mode structure and ignore relative mass across modes, yielding nearly identical scores for mixtures with different weights. Figure 1 illustrates this effect.

3 Learning energies via classification

In this work, we model the energy by jointly learning the time-dependent energy U_t^θ and log-normalizing constant $\mathrm{F}_t^\theta = -\log \mathcal{Z}_t^\theta$ [4], rather than focusing solely on the score. Our approach minimizes the DSM objective (7) together with a *Diffusive Classification* (`DiffCLF`) objective:

$$\mathcal{L}_{\mathrm{clf}}(\theta; N) = \mathbb{E}_{t_{1:N}}[\mathcal{L}_{\mathrm{clf}}(\theta; t_{1:N})], \quad \mathcal{L}_{\mathrm{clf}}(\theta; t_{1:N}) = -\frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{p_{t_i}}\left[\log \frac{p_{t_i}^\theta(Y_i)}{\sum_{j=1}^N p_{t_j}^\theta(Y_i)}\right], \quad (10)$$

[4] Note that F_t^θ is not the true negative log-normalizing constant, but only a learnable parameter.

where $p^\theta_{t_i}(y) = \exp(-\mathrm{U}^\theta_{t_i}(y) + \mathrm{F}^\theta_{t_i})$ and t_i are sampled independently and uniformly from $[0, T]$. It is noticeable that F^θ_t aims not normalizing p^θ_t but providing extra degree of freedom to aid training, which follows the same spirit as Gutmann & Hyvärinen (2010). In practice, F^θ_t is implemented as a bias on the last layer of the neural network U^θ_t. The objective (10) reformulates the task of estimating the EBM as a multi-class classification problem. Consider a sample y associated with a label c. If $c = i$, this indicates that y was generated from the marginal distribution at time t_i. In multinomial logistic regression, the goal is to estimate the posterior distribution over classes given the data y. Here, the class-conditional probability $p(\cdot|c = i)$ is modeled by $p^\theta_{t_i}$. If we further assume that all N labels are equally likely , then the posterior probabilities take the following form

$$p^\theta(c = i \mid y) = p^\theta_{t_i}(y) / \sum_{j=1}^{N} p^\theta_{t_j}(y) \ . \tag{11}$$

The categorical cross-entropy of this classifier corresponds exactly to Equation (10). Figure 2 illustrates this in the 3-level setting: given samples from three distinct time steps, the objective solves the classification problem to progressively separate them, as visualized by the posterior probabilities in the background. Because the classifier is constructed as a softmax over the EBMs (shown along the bottom edge of the figure), optimizing this objective also directly learns the marginal distribution of each sample group. As shown in the rightmost panel of Figure 1, the posterior probabilities (11) reflect changes in the mixture weights, unlike the score. This highlights that, in contrast to DSM, the classification objective doesn't suffer from mode blindness. Proposition 1 demonstrates that the true marginals attains the minimum of the `DiffCLF` objective.

Proposition 1. *The true marginals $(p_t)_{t\in[0,T]}$ are a minimizer of $\mathcal{L}_{\mathrm{clf}}$ (Equation* (10)*).*

However, the minimizer is not unique. For example, $p^\theta_t(y) = c(y)p_t(y)$ for any positive function c such that $\int c(y)p_t(y)dy = 1$ also attains the same minimum. To fix it, Proposition 2 shows that by optimizing the joint objective $\mathcal{L}_{\mathrm{DSM}} + \mathcal{L}_{\mathrm{clf}}$, the minimizer is unique.

Proposition 2 (Uniqueness - Informal)**.** *The unique minimizer for the joint objective $\mathcal{L}_{\mathrm{DSM}} + \mathcal{L}_{\mathrm{clf}}$ (Equations* (7) *and* (10)*) is attained by $p^{\theta^\star}_t = p_t$, for every $t \in [0, T]$.*

Furthermore, under mild regularity assumptions, the Monte-Carlo estimator of $\mathcal{L}_{\mathrm{DSM}} + \mathcal{L}_{\mathrm{clf}}$ (the quantity actually optimized during training) provably approximates the true objective. In particular, the empirical minimizer is both consistent and asymptotically normal:

Proposition 3 (Consistency - Informal)**.** *Let $\hat{\theta}^M$ denote the minimizer of the Monte-Carlo approximation of $\mathcal{L}_{\mathrm{DSM}} + \mathcal{L}_{\mathrm{clf}}$ based on M samples. Then:*

(i) $\hat{\theta}^M \xrightarrow{\mathbb{P}} \theta^\star$, where $\theta^\star$ is the minimizer of $\mathcal{L}_{\mathrm{DSM}} + \mathcal{L}_{\mathrm{clf}}$;

(ii) $\sqrt{M}(\hat{\theta}_M - \theta^\star)$ is asymptotically normal with mean zero and known covariance matrix Σ_N.

Moreover, Corollary 16 establishes that the norm of the asymptotic covariance Σ_N decreases as the number of classification levels N increases, reflecting the variance-reduction benefits of richer time-discretizations. Formal statements and proofs are provided in Appendix E.

In the simplest case of two times $t, t' \in [0, T]$, the objective reduces to binary classification, yielding

$$\mathcal{L}_{\mathrm{clf}}(\theta; t, t') = -\frac{1}{2}\mathbb{E}_{p_t}\left[\log h_\theta(Y_t; t, t')\right] - \frac{1}{2}\mathbb{E}_{p_{t'}}\left[\log(1 - h_\theta(Y_{t'}; t, t'))\right] \ , \tag{12}$$

where $h_\theta(y; t, t') = \mathrm{sigmoid}(-\mathrm{U}^\theta_t(y) + \mathrm{F}^\theta_t + \mathrm{U}^\theta_{t'}(y) - \mathrm{F}^\theta_{t'})$ and $\mathrm{sigmoid}(z) = 1/(1 + \exp(-z))$. In Appendix F, we generalize these objectives using Bergman divergences.

Computational cost. While the DSM objective (7) requires exactly two neural network evaluations per time sampled[5], the multi-class classification objective (10) requires N. Consequently, minimizing both $\mathcal{L}_{\mathrm{DSM}}$ and $\mathcal{L}_{\mathrm{clf}}$ simultaneously requires only $N + 1$ evaluations per time ($N - 1$ more evaluations than DSM). In the binary case, this amounts to just one additional evaluation compared to DSM, making the computational budgets highly comparable.

[5]One for the forward pass and one for the backward pass to compute the score with auto-differentiation.

4 CONNECTION TO OTHER WORKS

This section discusses connections between the proposed approach and existing methods. All the losses introduced, including `DiffCLF`, operate on the log-densities, without imposing a hard constraint to ensure the learned energies are normalized. Owing to space constraints, an extended related work section is provided in Appendix A, while additional connections are deferred to Appendix C.

4.1 CONSTRAINING TIME DERIVATIVES OF THE LOG-DENSITY

As highlighted in Section 2.3, regressing solely against the score is insufficient to recover the full energy landscape. Our approach tackles this challenge by using a loss that depends directly on the energy values, while alternative methods attempt to mitigate the same limitations through additional constraints on the model's higher-order derivatives.

Time Score Matching. An intuitive direction is to exploit the temporal structure of the process. This leads to optimizing EBMs so that their time-derivative aligns with that of the true marginals. We refer to this objective as *Time Score Matching* (tSM).

$$\mathcal{L}_{t\text{SM}}(\theta) = \mathbb{E}_t[\mathcal{L}_{t\text{SM}}(\theta; t)] \quad \text{with } \mathcal{L}_{t\text{SM}}(\theta; t) = \mathbb{E}_{p_t}\left[\left(\partial_t \log p_t^\theta(Y_t) - \partial_t \log p_t(Y_t)\right)^2\right] . \tag{13}$$

Figure 1 shows that, similar to the score, the time-score $\partial_t \log p_t$ also exhibits blindness of mode-weights. A theoretical justification is provided in Appendix D. Moreover, Proposition 5 establishes that the binary classification loss (12) converges to the tSM objective (17) in the continuous-time limit (see Appendix E.4 for the proof).

Proposition 4. *Let $t \in [0, T)$ and $\delta > 0$, we have*

$$\lim_{\delta \to 0^+} \frac{8}{\delta^2} \left(\mathcal{L}_{\text{clf}}(\theta; t, t+\delta) - \log 2\right) = \mathcal{L}_{t\text{SM}}(\theta; t) + C , \tag{14}$$

where $C = \mathbb{E}_{p_t}\left[\left(\frac{\partial}{\partial t} \log p_t(Y_t)\right)^2\right]$ is a constant with respect to θ.

Conditional Time Score Matching. In parallel, (Yu et al., 2025; Guth et al., 2025) propose an alternative objective, termed *Conditional Time Score Matching* (CtSM), which leverages the tractable conditional time score and enjoys the same gradients as the original formulation. For instance, in the DMs case where $X_t = S(t)X_0$, the time score can be expressed as $\partial \log p_t(y_t) = \mathbb{E}[\partial_t \log p_t(y_t|X_t)|Y_t = y_t]$ where the conditional time-score is

$$\partial_t \log p_t(Y_t|X_t) = -d\frac{\dot\gamma(t)}{\gamma(t)} - \frac{\partial_t X_t^\top (Y_t - X_t)}{\gamma(t)^2} + \frac{\|Y_t - X_t\|^2}{\gamma(t)^2}\frac{\dot\gamma(t)}{\gamma(t)} , \tag{15}$$

which leads to the following mean square regression problem

$$\mathcal{L}_{\text{C}t\text{SM}}(\theta) = \mathbb{E}_t[\mathcal{L}_{\text{C}t\text{SM}}(\theta; t)], \quad \mathcal{L}_{\text{C}t\text{SM}}(\theta; t) = \mathbb{E}\left[\left(\partial_t \log p_t^\theta(Y_t) - \partial_t \log p_t(Y_t|X_t)\right)^2\right] , \tag{16}$$

with $X_t \sim q_t$ and $Y_t \sim p_t(\cdot|X_t)$. This derivation closely parallels that of DSM (see Section 2.1). Similar to our approach, Yu et al. (2025) suggests combining this loss with DSM (7) to enhance model learning. Further, the conditional time score for SIs follows exactly the form in Equation (19); however, it remains intractable for general $(X_t)_t$ (see Appendix B for additional details).

4.2 CONNECTION TO NOISE CONTRASTIVE ESTIMATION

The binary objective (12) closely resembles the well-known *Noise Contrastive Estimation* (NCE) framework (Gutmann & Hyvärinen, 2010), with the multi-class extension (10) being related to the generalization considered in Matsuda & Hyvärinen (2019). Intuitively, our formulation can be interpreted as using the marginal at time t as the "noise distribution" when learning the density at t', and vice versa, but in a fully parametric setting. This connection highlights an additional flexibility of our approach: whenever some marginal densities p_t are known exactly (for example, p_T in diffusion models, or p_0 and p_1 in stochastic interpolants) these can be seamlessly incorporated into the learning framework, potentially improving accuracy.

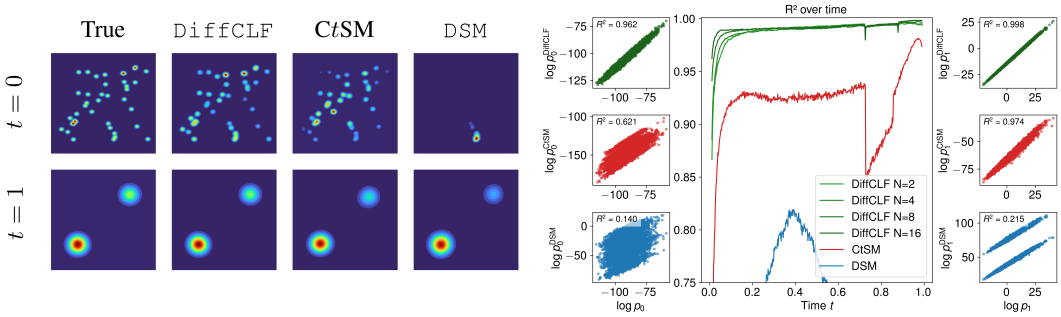
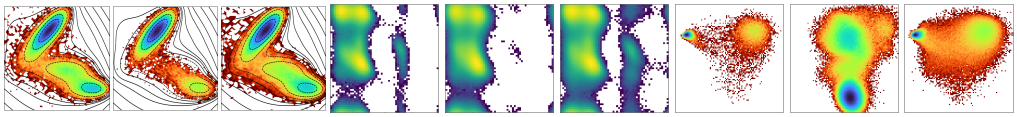


Figure 3: **Learned EBMs with SI between a bi-modal and a 40-mode Gaussian mixture.** We use $\mathcal{L}_{\mathrm{DSM}}$, $\mathcal{L}_{\mathrm{DSM}} + \mathcal{L}_{\mathrm{C}t\mathrm{SM}}$, and $\mathcal{L}_{\mathrm{DSM}} + \mathcal{L}_{\mathrm{clf}}$ (DiffCLF, ours). **(Left,** $d = 2$**)**: Learned densities at $t = 0$ (top row) and $t = 1$ (bottom row) for the different methods, showing that DiffCLF best captures the target distributions. **(Right,** $d = 128$**)**: Comparison of learned log-densities $\log p_t^\theta$ versus the exact $\log p_t$ on exact samples from $(Y_t)_t$ across time in terms of scatter plots and R^2 statistic. Plots at the left and right edges correspond to $t = 0$ and $t = 1$, respectively; the middle shows the coefficient of determination R^2 over $t \in (0, 1)$, indicating that only DiffCLF achieves consistently high agreement with the true log-densities.



(a) Müller-Brown (MB) (b) Alanine Dipeptide (ALDP) (c) Chignolin

Figure 4: Samples from Langevin dynamics with $\mathrm{U}_{t=0}^\theta$ on three benchmarks. **(Left)**: Reference; **(Middle)**: DSM; **(Right)**: DiffCLF. For MB we show the sample histogram; for ALDP, the torsion-angle histogram (x: ϕ, y: ψ); and for Chignolin, the histogram of the first two TIC axes.

5 Numerical Experiments

We begin our numerical study by comparing DiffCLF with DSM and CtSM [6] on controlled high-dimensional Gaussian mixtures, and then demonstrate its effectiveness and scalability on complex molecular systems, before turning to the practical applications in Section 2.2.

DMs and SIs on MOGs. In the mixture-of-Gaussians (MOG) setting, the closed-form density p_t is available (Appendix H.1), enabling quantitative evaluation of approximation errors. In Figure 3, SIs are trained to bridge a 40-mode (MOG-40) and a 2-mode (MOG-2) mixture in 2D and 128D, respectively, where DiffCLF learns substantially more accurate energies than all baselines. In Appendix H.2.3, we further train DMs on MOG-40 across increasing dimensions; results in Table 3 show that our method uniquely achieves low classification loss, satisfying the self-consistency condition missed by competing approaches. Additional results and details are provided in Appendix H.2.

Evaluation on molecular systems. We further conduct experiments on training energy-based DMs for molecular systems, including Müller-Brown potential, Alanine Dipeptide, and Chignolin. We train DMs with $\mathcal{L}_{\mathrm{DSM}}$ only and $\mathcal{L}_{\mathrm{DSM}} + \mathcal{L}_{\mathrm{clf}}$, then evaluate the learned energy by running molecular dynamics using $\mathrm{U}_{t=0}^\theta$, where we follow the same settings as Plainer et al. (2025). Figure 4 and its full version in Figure 10 demonstrate the effectiveness and scalability of DiffCLF, which aids better learning of energies while not degenerating diffusion sampling. Details of experiment can be found in Appendix H.6.

Composition. Following Du et al. (2023); Thornton et al. (2025), we compare DiffCLF with DSM on two toy composition tasks (Figure 5, left): an "OR" of two Gaussian mixtures with different weights and an "AND" of rings and uniform rectangles. Composition is performed via SMC (Doucet et al., 2001; Del Moral et al., 2006) with a Metropolized Langevin (MALA) kernel applied

[6] For clarity, DiffCLF refers to $\mathcal{L}_{\mathrm{DSM}} + \mathcal{L}_{\mathrm{clf}}$ training and CtSM refers $\mathcal{L}_{\mathrm{DSM}} + \mathcal{L}_{\mathrm{C}t\mathrm{SM}}$ training.

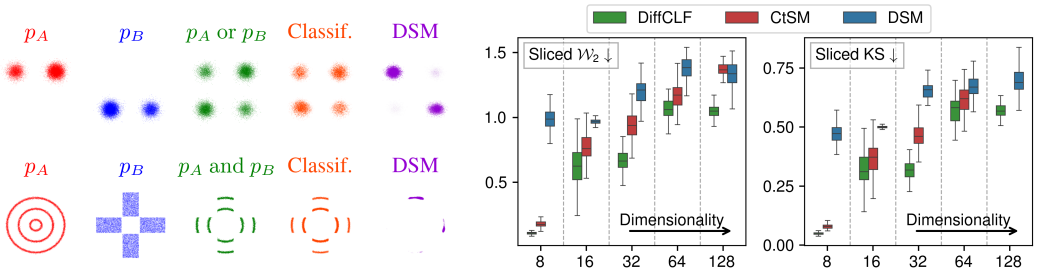


Figure 5: **(Left) OR and AND model composition.** *Top:* OR composition, *Bottom:* AND composition. Red/Blue: input distributions, Green: ground truth, Orange: `DiffCLF`, Purple: DSM. Results obtained via 512-step SMC on the product of learned marginals. **(Right) SMC-based BG metrics.** Box plots of Sliced Wasserstein ($\mathcal{W}_2$) and Kolmogorov-Smirnov (KS) distances for a 512-step SMC on the SI between MOG-40 and MOG-2. Optimal scores and velocities are used for kernels, with learned EBMs for marginals. `DiffCLF` consistently outperforms other methods.

to mixtures or products of learned densities. As shown in the two rightmost columns, `DiffCLF` substantially improves over DSM, especially in preserving correct region proportions, by avoiding mode blindness. Details are in Appendix H.3.

Boltzmann Generators. To demonstrate the use of trained EBMs for constructing Boltzmann Generators, we consider an energy-based SI bridging MOG-40 and MOG-2, embedding the learned energy into the SMC framework of Phillips et al. (2024), which integrates the SDE (9) to improve inter-level transitions. The algorithm uses energies learned by `DiffCLF`, C*t*SM, or DSM, with the final marginal constrained post hoc to match the true distribution. To isolate the effect of energy estimation, we employ analytical velocities and scores (Appendix H.1) to compute transition densities. As shown in Figure 5 (right), `DiffCLF` produces substantially more accurate samples than DSM and C*t*SM across all dimensions, underscoring its advantage for BG construction.

Free energy difference estimation. We train SIs to enhance free-energy difference estimation via TI. The experiments focus on the estimating free energy difference of alanine dipeptide in implicit solvent (ALDP-imp) and in vacuum (ALDP-vac) at temperature $T = 300$K. Models are trained using $\mathcal{L}_{\text{base}} + \mathcal{L}_{\text{clf}}$, where $\mathcal{L}_{\text{base}}$ denotes the objective introduced in Máté et al. (2025). The resulting estimates [7] are reported in Table 1. The estimate produced by `DiffCLF` exhibits improved accuracy.

Table 1: ALDP solvation free energy estimated with thermodynamic integration.

Method	Estimation
Reference	$29.43_{\pm 0.01}$
TI w/ $\mathcal{L}_{\text{base}}$ (Máté et al., 2025)	$27.30_{\pm 0.45}$
TI w/ $\mathcal{L}_{\text{base}} + \mathcal{L}_{\text{clf}}$ (ours)	$\mathbf{29.02}_{\pm 0.41}$

6 Conclusion

Energy-based generative models provide a compelling extension of score-based methods, enabling applications beyond pure sample generation. However, existing energy-based training approaches are limited, often being computationally costly, hyperparameter-sensitive, or biased. We introduce the *Diffusive Classification* (`DiffCLF`) objective, a simple, efficient, and unbiased training principle. The method is broadly applicable across stochastic processes and integrates naturally with existing approaches such as DSM, offering theoretical clarity and practical flexibility. Empirically, `DiffCLF` yields more accurate and consistent energy estimates, improving both model composition and model-induced Boltzmann Generators. Nonetheless, our experiments remain limited in scale. Applying `DiffCLF` to large-scale problems such as image modeling—where SMC-based composition has shown promise (Thornton et al., 2025)—is a natural next step. A further extension to discrete domains is also appealing: since `DiffCLF` applies to general stochastic processes, including continuous-time Markov chains (see Appendix G for more details), its use in text modeling aligns well with recent progress connecting EBMs and DMs (Xu et al., 2025).

[7] The reference value is from He et al. (2025a).

LLM USAGE DISCLOSURE

LLM was used at the sentence level to correct grammar and improve clarity.

ACKNOWLEDGEMENTS

The authors would like to thank Omar Chehab, Jiajun He, and Hanlin Yu, for the helpful discussions and feedbacks. LG acknowledges government funding managed by the French National Research Agency under France 2030, reference ANR-23-IACL-0005. RKOY acknowledges the UK Engineering and Physical Sciences Research Council (EPSRC) grant EP/L016516/1 for the University of Cambridge Centre for Doctoral Training, the Cambridge Centre for Analysis. JMHL acknowledges support from EPSRC funding under grant EP/Y028805/1. JMHL also acknowledges support from a Turing AI Fellowship under grant EP/V023756/1. This work was performed using HPC resources from GENCI–IDRIS (AD011015234R1). This project acknowledges the resources provided by the Cambridge Service for Data-Driven Discovery (CSD3) operated by the University of Cambridge Research Computing Service (www.csd3.cam.ac.uk), provided by Dell EMC and Intel using Tier-2 funding from the Engineering and Physical Sciences Research Council (capital grant EP/T022159/1), and DiRAC funding from the Science and Technology Facilities Council (www.dirac.ac.uk).

REFERENCES

S. Agapiou, O. Papaspiliopoulos, D. Sanz-Alonso, and A. M. Stuart. Importance sampling: Intrinsic dimension and computational cost. *Statistical Science*, 32(3):405–431, 2017. ISSN 08834237, 21688745. URL http://www.jstor.org/stable/26408299.

Rishal Aggarwal, Jacky Chen, Nicholas M. Boffi, and David Ryan Koes. Boltznce: Learning likelihoods for boltzmann generation with stochastic interpolants and noise contrastive estimation, 2025. URL https://arxiv.org/abs/2507.00846.

Michael S. Albergo, Nicholas M. Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions, 2023. URL https://arxiv.org/abs/2303.08797.

Michael Samuel Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=li7qeBbCR1t.

Brian D.O. Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982. ISSN 0304-4149. doi: https://doi.org/10.1016/0304-4149(82)90051-5. URL https://www.sciencedirect.com/science/article/pii/0304414982900515.

C. H. Bennett. Efficient Estimation of Free Energy Differences from Monte Carlo Data. *Journal of Computational Physics*, 22(2):245–268, October 1976. doi: 10.1016/0021-9991(76)90078-4.

Valentin De Bortoli, Michael Hutchinson, Peter Wirnsberger, and Arnaud Doucet. Target score matching, 2024. URL https://arxiv.org/abs/2402.08667.

Lev M Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR computational mathematics and mathematical physics*, 7(3):200–217, 1967.

Andrew Campbell, Joe Benton, Valentin De Bortoli, Thomas Rainforth, George Deligiannidis, and Arnaud Doucet. A continuous time framework for discrete denoising models. *Advances in Neural Information Processing Systems*, 35:28266–28279, 2022.

Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver,

Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 5453–5512. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/campbell24a.html.

Davide Carbone, Mengjian Hua, Simon Coste, and Eric Vanden-Eijnden. Efficient training of energy-based models using jarzynski equality. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=MXxZ0Z5MNz.

Tong Che, Ruixiang ZHANG, Jascha Sohl-Dickstein, Hugo Larochelle, Liam Paull, Yuan Cao, and Yoshua Bengio. Your gan is secretly an energy-based model and you should use discriminator driven latent sampling. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 12275–12287. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/90525e70b7842930586545c6f1c9310c-Paper.pdf.

John Chodera, Andrea Rizzi, Levi Naden, Kyle Beauchamp, Patrick Grinaway, Mike Henry, Iván Pulido, Josh Fass, Alex Wade, Gregory A. Ross, Andreas Kraemer, Hannah Bruce Macdonald, jaimergp, Bas Rustenburg, David W.H. Swenson, Ivy Zhang, Dominic Rufa, Andy Simmonett, Mark J. Williamson, hb0402, Jake Fennick, Sander Roet, Benjamin Ries, Ian Kenney, Irfan Alibay, Richard Gowers, and SimonBoothroyd. choderalab/openmmtools: 0.24.1, January 2025. URL https://doi.org/10.5281/zenodo.14782825.

Kristy Choi, Chenlin Meng, Yang Song, and Stefano Ermon. Density ratio estimation via infinitesimal classification. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera (eds.), *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 2552–2573. PMLR, 28–30 Mar 2022. URL https://proceedings.mlr.press/v151/choi22a.html.

Nicolas Chopin and Omiros Papaspiliopoulos. *An introduction to sequential Monte Carlo*. Springer International Publishing, Cham, 2020. ISBN 978-3-030-47845-2. doi: 10.1007/978-3-030-47845-2_8. URL https://doi.org/10.1007/978-3-030-47845-2_8.

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(3):411–436, 2006.

Arnaud Doucet, Nando De Freitas, Neil James Gordon, et al. *Sequential Monte Carlo methods in practice*, volume 1. Springer, 2001.

Yilun Du and Igor Mordatch. Implicit generation and modeling with energy based models. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/378a063b8fdb1db941e34f4bde584c7d-Paper.pdf.

Yilun Du, Shuang Li, Joshua Tenenbaum, and Igor Mordatch. Improved contrastive divergence training of energy-based models. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 2837–2848. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/du21b.html.

Yilun Du, Conor Durkan, Robin Strudel, Joshua B. Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Sussman Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and MCMC. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 8489–8510. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/du23a.html.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011. ISSN 01621459. URL http://www.jstor.org/stable/23239562.

Ruiqi Gao, Yang Song, Ben Poole, Ying Nian Wu, and Diederik P Kingma. Learning energy-based models by diffusion recovery likelihood. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=v_1Soh8QUNc.

Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky T. Q. Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete flow matching. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 133345–133385. Curran Associates, Inc., 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/f0d629a734b56a642701bba7bc8bb3ed-Paper-Conference.pdf.

Will Grathwohl, Kuan-Chieh Wang, Joern-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your classifier is secretly an energy based model and you should treat it like one. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=Hkxzx0NtDB.

Will Grathwohl, Jacob Kelly, Milad Hashemi, Mohammad Norouzi, Kevin Swersky, and David Duvenaud. No mcmc for me: Amortized sampling for fast and stable training of energy-based models, 2021. URL https://arxiv.org/abs/2010.04230.

Louis Grenioux, Maxence Noble, Marylou Gabrié, and Alain Oliviero Durmus. Stochastic localization via iterative posterior sampling. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 16337–16376. PMLR, 2024. URL https://proceedings.mlr.press/v235/grenioux24a.html.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012. URL http://jmlr.org/papers/v13/gretton12a.html.

Florentin Guth, Zahra Kadkhodaie, and Eero P Simoncelli. Learning normalized image densities via dual score matching, 2025. URL https://arxiv.org/abs/2506.05310.

Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In Yee Whye Teh and Mike Titterington (eds.), *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pp. 297–304, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR. URL https://proceedings.mlr.press/v9/gutmann10a.html.

Michael Gutmann and Jun ichiro Hirayama. Bregman divergence as general framework to estimate unnormalized statistical models, 2012. URL https://arxiv.org/abs/1202.3727.

Jiajun He, Yuanqi Du, Francisco Vargas, Yuanqing Wang, Carla P. Gomes, José Miguel Hernández-Lobato, and Eric Vanden-Eijnden. Feat: Free energy estimators with adaptive transport, 2025a. URL https://arxiv.org/abs/2504.11516.

Jiajun He, José Miguel Hernández-Lobato, Yuanqi Du, and Francisco Vargas. Rne: plug-and-play diffusion inference-time control and energy-based training, 2025b. URL https://arxiv.org/abs/2506.05668.

Jiajun He, Paul Jeha, Peter Potaptchik, Leo Zhang, José Miguel Hernández-Lobato, Yuanqi Du, Saifuddin Syed, and Francisco Vargas. Crepe: Controlling diffusion with replica exchange, 2025c. URL https://arxiv.org/abs/2509.23265.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005. URL http://jmlr.org/papers/v6/hyvarinen05a.html.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 26565–26577. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/a98846e9d9cc01cfb87eb694d946ce6b-Paper-Conference.pdf.

Frank P Kelly. *Reversibility and stochastic networks*. Cambridge University Press, 2011.

Taesup Kim and Yoshua Bengio. Deep directed generative models with energy-based probability estimation. In *Fifth International Conference on Machine Learning*, 2016. URL https://openreview.net/forum?id=BNYAGZZj5S7PwR1riXzA.

Diederik P Kingma and Ruiqi Gao. Understanding diffusion objectives as the ELBO with simple data augmentation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=NnMEadcdyD.

John G Kirkwood. Statistical mechanics of fluid mixtures. *The Journal of chemical physics*, 3(5): 300–313, 1935.

Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and F Huang. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

Hankook Lee, Jongheon Jeong, Sejun Park, and Jinwoo Shin. Guiding energy-based models via contrastive latent variables. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=CZmHHj9MgkP.

Tony Lelièvre, Mathias Rousset, and Gabriel Stoltz. *Free Energy Computations*. IMPERIAL COLLEGE PRESS, 2010. doi: 10.1142/p579. URL https://www.worldscientific.com/doi/abs/10.1142/p579.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. *arXiv preprint arXiv:2310.16834*, 2023.

Cheng Lu, Kaiwen Zheng, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Maximum likelihood training for score-based diffusion ODEs by high order denoising score matching. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 14429–14460. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/lu22f.html.

Takeru Matsuda and Aapo Hyvärinen. Estimation of non-normalized mixture models. In Kamalika Chaudhuri and Masashi Sugiyama (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 2555–2563. PMLR, 16–18 Apr 2019. URL https://proceedings.mlr.press/v89/matsuda19a.html.

Chenlin Meng, Kristy Choi, Jiaming Song, and Stefano Ermon. Concrete score matching: Generalized score matching for discrete data, 2023. URL https://arxiv.org/abs/2211.00802.

Laurence Illing Midgley, Vincent Stimper, Gregor N. C. Simm, Bernhard Schölkopf, and José Miguel Hernández-Lobato. Flow annealed importance sampling bootstrap. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=XCTVFJwS9LJ.

Bálint Máté, François Fleuret, and Tristan Bereau. Solvation free energies from neural thermodynamic integration. *The Journal of Chemical Physics*, 162(12):124107, 03 2025. ISSN 0021-9606. doi: 10.1063/5.0251736. URL https://doi.org/10.1063/5.0251736.

Radford M Neal. Annealed importance sampling. *Statistics and computing*, 11:125–139, 2001.

Erik Nijkamp, Mitch Hill, Song-Chun Zhu, and Ying Nian Wu. On learning non-convergent short-run mcmc toward energy-based model. *arXiv preprint arXiv:1904.09770*, 2019.

Maxence Noble, Louis Grenioux, Marylou Gabrié, and Alain Oliviero Durmus. Learned reference-based diffusion sampler for multi-modal distributions. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=fmJUYgmMbL.

Frank Noé, Simon Olsson, Jonas Köhler, and Hao Wu. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147, 9 2019. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.aaw1147. URL https://www.science.org/doi/10.1126/science.aaw1147.

Angus Phillips, Hai-Dang Dau, Michael John Hutchinson, Valentin De Bortoli, George Deligiannidis, and Arnaud Doucet. Particle denoising diffusion sampler. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 40688–40724. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/phillips24a.html.

Michael Plainer, Hao Wu, Leon Klein, Stephan Günnemann, and Frank Noé. Consistent sampling and simulation: Molecular dynamics with energy-based diffusion models, 2025. URL https://arxiv.org/abs/2506.17139.

Gareth O. Roberts and Richard L. Tweedie. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363, 1996. ISSN 13507265. URL http://www.jstor.org/stable/3318418.

Tim Salimans and Jonathan Ho. Should EBMs model the energy or the score? In *Energy Based Models Workshop - ICLR 2021*, 2021. URL https://openreview.net/forum?id=9AS-TF2jRNb.

Tobias Schröder, Zijing Ou, Jen Lim, Yingzhen Li, Sebastian Vollmer, and Andrew Duncan. Energy discrepancies: A score-independent loss for energy-based models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 45300–45338. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/8e176ef071f00f1b233461c5ad5e1b24-Paper-Conference.pdf.

Hugo Senetaire, Paul Jeha, Pierre-Alexandre Mattei, and Jes Frellsen. Learning energy-based models by self-normalising the likelihood, 2025. URL https://arxiv.org/abs/2503.07021.

Neta Shaul, Itai Gat, Marton Havasi, Daniel Severo, Anuroop Sriram, Peter Holderrieth, Brian Karrer, Yaron Lipman, and Ricky T. Q. Chen. Flow matching with general discrete paths: A kinetic-optimal perspective. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=tcvMzR2NrP.

Zhekun Shi, Longlin Yu, Tianyu Xie, and Cheng Zhang. Diffusion-PINN sampler, 2024. URL https://arxiv.org/abs/2410.15336.

Michael R. Shirts and John D. Chodera. Statistically optimal analysis of samples from multiple equilibrium states. *The Journal of Chemical Physics*, 129(12), September 2008. ISSN 1089-7690. doi: 10.1063/1.2978177. URL http://dx.doi.org/10.1063/1.2978177.

Marta Skreta, Tara Akhound-Sadegh, Viktor Ohanesian, Roberto Bondesan, Alan Aspuru-Guzik, Arnaud Doucet, Rob Brekelmans, Alexander Tong, and Kirill Neklyudov. Feynman-kac correctors in diffusion: Annealing, guidance, and product of experts. In *Forty-second International Conference on Machine Learning*, 2025a. URL https://openreview.net/forum?id=Vhc0KrcqWu.

Marta Skreta, Lazar Atanackovic, Joey Bose, Alexander Tong, and Kirill Neklyudov. The superposition of diffusion models using the itô density estimator. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=2o58Mbqkd2.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 2256–2265, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/sohl-dickstein15.html.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=St1giarCHLP.

Yang Song and Diederik P. Kingma. How to train your energy-based models, 2021. URL https://arxiv.org/abs/2101.03288.

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 (ICLR)*, 2021b.

Jingtong Sun, Julius Berner, Lorenz Richter, Marius Zeinhofer, Johannes Müller, and Kamyar Azizzadenesheli andf Anima Anandkumar. Dynamical measure transport and neural pde solvers for sampling, 2024. URL https://arxiv.org/abs/2407.07873.

James Thornton, Louis Béthune, Ruixiang ZHANG, Arwen Bradley, Preetum Nakkiran, and Shuangfei Zhai. Controlled generation with distilled diffusion energy models and sequential monte carlo. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2025. URL https://openreview.net/forum?id=6GyX0YRw8P.

Li K. Wenliang and Heishiro Kanagawa. Blindness of score-based methods to isolated components and mixing proportions, 2021. URL https://arxiv.org/abs/2008.10087.

Minkai Xu, Tomas Geffner, Karsten Kreis, Weili Nie, Yilun Xu, Jure Leskovec, Stefano Ermon, and Arash Vahdat. Energy-based diffusion language models for text generation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=sL2F9YCMXf.

Hanlin Yu, Arto Klami, Aapo Hyvarinen, Anna Korba, and Omar Chehab. Density ratio estimation with conditional probability paths. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=Gn2izAiYzZ.

Fengzhe Zhang, Jiajun He, Laurence I Midgley, Javier Antorán, and José Miguel Hernández-Lobato. Efficient and unbiased sampling of boltzmann distributions via consistency models. *arXiv preprint arXiv:2409.07323*, 2024.

Fengzhe Zhang, Laurence I. Midgley, and José Miguel Hernández-Lobato. Efficient and unbiased sampling from boltzmann distributions via variance-tuned diffusion models, 2025a. URL https://arxiv.org/abs/2505.21005.

Leo Zhang, Peter Potaptchik, Jiajun He, Yuanqi Du, Arnaud Doucet, Francisco Vargas, Hai-Dang Dau, and Saifuddin Syed. Accelerated parallel tempering via neural transports, 2025b. URL https://arxiv.org/abs/2502.10328.

Mingtian Zhang, Oscar Key, Peter Hayes, David Barber, Brooks Paige, and Francois-Xavier Briol. Towards healing the blindness of score matching. In *NeurIPS 2022 Workshop on Score-Based Methods*, 2022. URL https://openreview.net/forum?id=Ij8G_k0iuL.

Xinwei Zhang, Zhiqiang Tan, and Zhijian Ou. Persistently trained, diffusion-assisted energy-based models. *Stat*, 12(1):e625, 2023. doi: https://doi.org/10.1002/sta4.625. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/sta4.625.

Yaxuan Zhu, Jianwen Xie, Ying Nian Wu, and Ruiqi Gao. Learning energy-based models by cooperative diffusion recovery likelihood. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=AyzkDpuqcl.

Robert W Zwanzig. High-temperature equation of state by a perturbation method. i. nonpolar gases. *The Journal of Chemical Physics*, 22(8):1420–1426, 1954.

A Diffusive Classification Loss for Learning Energy-based Generative Models Appendix

A Extended Related Works

This section discusses connections between the proposed approach and existing methods. It first focuses on approaches that constrain higher-order derivatives of the log-density (Appendix A.1), and then considers works that directly operate on the log-density itself (Appendix A.2). All the losses introduced, including `DiffCLF`, operate on the log-densities, without imposing a hard constraint to ensure the learned energies are normalized.

A.1 Constraining other derivatives of the log-density

As highlighted in Section 2.3, regressing solely against the score is insufficient to recover the full energy landscape. Our approach tackles this challenge by using a loss that depends directly on the energy values, while alternative methods attempt to mitigate the same limitations through additional constraints on the model's higher-order derivatives.

A.1.1 Connection to regressing time derivatives of log-densities.

Time Score Matching. An intuitive direction is to exploit the temporal structure of the process. This leads to optimizing EBMs so that their time-derivative aligns with that of the true marginals. We refer to this objective as *Time Score Matching* (tSM).

$$\mathcal{L}_{t\text{SM}}(\theta) = \mathbb{E}_t[\mathcal{L}_{t\text{SM}}(\theta; t)] \quad \text{with } \mathcal{L}_{t\text{SM}}(\theta; t) = \mathbb{E}_{p_t}\left[\left(\partial_t \log p_t^\theta(Y_t) - \partial_t \log p_t(Y_t)\right)^2\right] . \tag{17}$$

Figure 1 shows that, similar to the score, the time-score $\partial_t \log p_t$ also exhibits blindness of mode-weights. A theoretical justification is provided in Appendix D. Moreover, Proposition 5 establishes that the binary classification loss (12) converges to the tSM objective (17) in the continuous-time limit (see Appendix E.4 for the proof).

Proposition 5. *Let $t \in [0, T)$ and $\delta > 0$, we have*

$$\lim_{\delta \to 0^+} \frac{8}{\delta^2} \left(\mathcal{L}_{\text{clf}}(\theta; t, t+\delta) - \log 2\right) = \mathcal{L}_{t\text{SM}}(\theta; t) + C , \tag{18}$$

where $C = \mathbb{E}_{p_t}\left[\left(\frac{\partial}{\partial t} \log p_t(Y_t)\right)^2\right]$ is a constant with respect to θ.

DRE-∞. However, as with the score, directly regressing the time-score is generally infeasible as it is usually intractable. (Choi et al., 2022, Proposition 4) extends Proposition 5 by showing that the optimal binary classifier associated with objective (12) (using a general classifier) can be used to approximate the time-score. In contrast, the proposed approach embeds the parameterized marginals directly into the classification problem, bypassing the need to approximate the time-score altogether.

Conditional Time Score Matching. In parallel, (Yu et al., 2025; Guth et al., 2025) propose an alternative objective, termed *Conditional Time Score Matching* (CtSM), which leverages the tractable conditional time score and enjoys the same gradients as the original formulation. For instance, in the DMs case where $X_t = S(t)X_0$, the time score can be expressed as $\partial \log p_t(y_t) = \mathbb{E}[\partial_t \log p_t(y_t|X_t)|Y_t = y_t]$ where the conditional time-score is

$$\partial_t \log p_t(Y_t|X_t) = -d\frac{\dot\gamma(t)}{\gamma(t)} - \frac{\partial_t X_t^\top (Y_t - X_t)}{\gamma(t)^2} + \frac{\|Y_t - X_t\|^2}{\gamma(t)^2}\frac{\dot\gamma(t)}{\gamma(t)} , \tag{19}$$

which leads to the following mean square regression problem

$$\mathcal{L}_{\text{C}t\text{SM}}(\theta) = \mathbb{E}_t[\mathcal{L}_{\text{C}t\text{SM}}(\theta; t)], \quad \mathcal{L}_{\text{C}t\text{SM}}(\theta; t) = \mathbb{E}\left[\left(\partial_t \log p_t^\theta(Y_t) - \partial_t \log p_t(Y_t|X_t)\right)^2\right] , \tag{20}$$

with $X_t \sim q_t$ and $Y_t \sim p_t(\cdot|X_t)$. This derivation closely parallels that of DSM (see Section 2.1). Similar to our approach, Choi et al. (2022) and Yu et al. (2025) suggest combining this loss with DSM (7) to enhance model learning. Further, the conditional time score for SIs follows exactly the form in Equation (19); however, it remains intractable for general $(X_t)_t$ (see Appendix B for the proof and additional details).

A.1.2 Learning Log-densities with Self-consistency.

Another strategy for estimating log-densities is to enforce *self-consistency* relations implied by the dynamics of $(Y_t)_t$. Two main approaches have been explored: one derived from the *Fokker–Planck equation* (FPE) and another from *Bayes' rule*. These methods typically require stronger assumptions on the process and are best understood in structured settings such as DMs or SIs. The detailed introduction and discussion are provided in appendix B.4.

Consistency via Fokker–Planck. When $(Y_t)_t$ admits an SDE representation, its marginals follow the FPE, a partial differential equation governing the log-densities $(\log p_t)_t$. Several works (Sun et al., 2024; Shi et al., 2024) exploit this by penalizing deviations from the log-density FPE. While conceptually appealing, this requires backpropagating through time-derivatives, scores, and Laplacians, leading to high computational cost. Variants mitigate these issues by approximating derivatives with finite differences (Plainer et al., 2025). Moreover, Appendix D shows that such objectives remain subject to mode blindness, despite claims to the contrary.

Consistency via Bayes' Rule. An alternative derives from the relation between marginals at two times s and t, $p_t(y_t)\, p_{s|t}(y_s|y_t) = p_s(y_s)\, p_{t|s}(y_t|y_s)$ for all $y_s, y_t \in \mathbb{R}^d$, where $p_{t|s}$ (resp. $p_{s|t}$) is the conditional distribution of Y_t given $Y_s = y_s$ (resp. Y_s given $Y_t = y_t$). Using approximations of the conditional distributions given by Euler–Maruyama integration of SDEs (8) or (9) (which depend on the score), one can regularize $(\log p_t^\theta)_t$ by enforcing approximate Bayes consistency (He et al., 2025b). However, this method remains valid only when the approximations are sufficiently accurate, which occurs for s, t close together, precisely the regime where the objective is prone to mode blindness. In fact, Proposition 6 in Appendix B.4 shows that the Bayes and FPE regularizations coincide asymptotically, inheriting the same limitations.

A.2 Connection to other training methods

Maximum likelihood approaches. While direct ML on a single distribution (3) is notoriously difficult, temporal correlations in $(Y_t)_t$ can alleviate this. Noble et al. (2025) learn an annealing path $(p_t^\theta)t$, while Zhang et al. (2023) model the joint time–state distribution and use Gibbs transitions across levels. In the DMs case, Gao et al. (2021); Zhu et al. (2024) exploit tractable conditionals $p_{t|s}$ to model posteriors as EBM with $p^\theta_{s|t} \propto p_{t|s} \times p_t^\theta$, enabling more efficient ML when $t - s \to 0$. Despite their advantages, these methods still depend on costly sampling loops.

Noise contrastive estimation. The binary objective (12) closely resembles the well-known *Noise Contrastive Estimation* (NCE) framework (Gutmann & Hyvärinen, 2010), with the multi-class extension (10) being related to the generalization considered in Matsuda & Hyvärinen (2019). Intuitively, our formulation can be interpreted as using the marginal at time t as the "noise distribution" when learning the density at t', and vice versa, but in a fully parametric setting. This connection highlights an additional flexibility of our approach: whenever some marginal densities p_t are known exactly (for example, p_T in diffusion models, or p_0 and p_1 in stochastic interpolants) these can be seamlessly incorporated into the learning framework, potentially improving accuracy. More recently, Aggarwal et al. (2025) also considers using NCE to distill model density of Diffusion models.

B Additional background

B.1 Background and Foundations for Itô SDEs

In this section, we will introduce the *Itô SDEs*, its *time reversal*, and the *Fokker Planck Equation*.

Forward and Backward Processes of Itô SDEs. Given $f : [0,T] \times \mathbb{R}^d \to \mathbb{R}^d$ and $g : [0,T] \times \mathbb{R}^d \to \mathbb{R}^d$ with regularity, the Itô SDE is defined as

$$\mathrm{d}X_t = f(t, X_t)\mathrm{d}t + g(t, X_t)\mathrm{d}W_t, \quad t \in [0,T]\,, \tag{21}$$

where $(W_t)_{t\in[0,T]}$ is the standard Wiener process. With additional regularity on f and g (see Anderson (1982)), its *time reversal* is given as

$$\mathrm{d}X_t = \left[f(t, X_t) - g(t, X_t)g(t, X_t)^\top \nabla \log p_t(X_t)\right] \mathrm{d}t + g(t, X_t)\mathrm{d}\tilde{W}_t\,, \tag{22}$$

where $(\tilde{W}_t)_{t\in[0,T]}$ is the time-reversed Wiener process. We denote $(p_t)_{t\in[0,T]}$ as the marginal distributions admitted by the SDE, starting either from $(t=0, X_0 \sim p_0)$ or $(t=T, X_T \sim p_T)$. For simplicity, in the rest of context, we consider the *Additive-noise Itô SDEs*, *i.e.* with $g(t,x)=g(t)$, which gives

$$\text{(forward SDE)} \quad \mathrm{d}X_t = f(t,X_t)\mathrm{d}t + g(t)\mathrm{d}W_t \ , \tag{23}$$

$$\text{(backward SDE)} \quad \mathrm{d}X_t = \left[f(t,X_t) - g(t)^2\nabla \log p_t(X_t)\right]\mathrm{d}t + g(t)\mathrm{d}\tilde{W}_t \ . \tag{24}$$

Fokker Planck Equation. The *Fokker Planck Equation* (FPE), which is also known as the *Forward Kolmogorov Equation*, describes the density change along the forward SDE as follows

$$\partial_t p_t(x) = -\nabla\cdot(f(t,x)p_t(x)) + \frac{g(t)^2}{2}\Delta\cdot p_t(x) \ , \tag{25}$$

where ∇ is the divergence operator and Δ is the Laplacian operator both with respect to the variable x. Its log-density version, *i.e. log-density FPE* is given by

$$\partial_t \log p_t(x) = -\nabla\cdot f(x,t) - f(x,t)^\top\nabla\log p_t(x) + \frac{g(t)^2}{2}\left[\nabla\cdot\nabla\log p_t(x) + \|\nabla\log p_t(x)\|^2\right] \ . \tag{26}$$

B.2 On the practical parameterization of noising processes of DMs

In this part, we will introduce the parameterization of noising processes in DMs, which is introduced by Karras et al. (2022). Recall the noising SDE defined as

$$\mathrm{d}Y_t = f(t)Y_t\mathrm{d}t + g(t)\mathrm{d}W_t \ ,$$

where $(W_t)_t$ is a standard Brownian motion. The noising kernel, $p_{t|0}(y_t|y_0)$, is a Gaussian distribution with closed-form expressions of the mean and covariance due to the linearity of drift (Song et al., 2021b; Karras et al., 2022):

$$p_{t|0}(y_t|y_0) = \mathcal{N}(y_t; S(t)y_0, \Sigma(t)) \ ,$$

where

$$S(t) = \exp\left(\int_0^t f(u)\mathrm{d}u\right), \quad \Sigma(t) = \left[S(t)^2\int_0^t \frac{g(u)^2}{S(u)^2}\mathrm{d}u\right]\mathrm{I}_d \ . \tag{27}$$

Let $\sigma(t)^2 = \int_0^t (g(u)^2/S(u)^2)\mathrm{d}u$, we have

$$\dot{S}(t) = \frac{\mathrm{d}}{\mathrm{d}t}S(t) = \frac{\mathrm{d}}{\mathrm{d}t}\exp\left(\int_0^t f(u)\mathrm{d}u\right) = S(t)f(t) \Longrightarrow f(t) = \frac{\dot{S}(t)}{S(t)} \ , \tag{28}$$

$$\dot{\sigma}(t) = \frac{\mathrm{d}}{\mathrm{d}t}\sigma(t) = \frac{\mathrm{d}}{\mathrm{d}t}\sqrt{\int_0^t \frac{g(u)^2}{S(u)^2}\mathrm{d}u} = \frac{1}{2\sigma(t)}\frac{g(t)^2}{S(t)^2} \Longrightarrow g(t) = S(t)\sqrt{2\dot{\sigma}(t)\sigma(t)} \ . \tag{29}$$

Variance Preserving. The *Variance Preserving* (VP) SDE (Song et al., 2021b) is defined as

$$f(t) = -\frac{1}{2}\beta(t), \quad g(t) = \sqrt{\beta(t)}, \quad \text{and } \beta(t) = \beta_{\min} + t(\beta_{\max}-\beta_{\min}) \ ,$$

that is,

$$S(t) = \exp\left(\int_0^t -\frac{1}{2}\beta(u)\mathrm{d}u\right) = \exp\left(-\frac{1}{2}\left(\frac{\beta_{\max}-\beta_{\min}}{2}t^2 + \beta_{\min}t\right)\right) \ ,$$

$$\sigma(t)^2 = \int_0^t \frac{\beta_{\min}+(\beta_{\max}-\beta_{\min})u}{\exp\left(-\frac{\beta_{\max}-\beta_{\min}}{2}u^2 - \beta_{\min}u\right)}\mathrm{d}u = \exp\left(\frac{1}{2}\left(\frac{\beta_{\max}-\beta_{\min}}{2}t^2 + \beta_{\min}t\right)\right) - 1 \ .$$

Variance Exploding. The *variance Exploding* (VE) SDE (Song et al., 2021b) is defined as

$$f(t) \equiv 0, \quad g(t) = \sigma_{\min}\sqrt{2\log\sigma_{\mathrm{d}}}\sigma_{\mathrm{d}}^{t}, \quad \text{and } \sigma_{\mathrm{d}} = \sigma_{\max}/\sigma_{\min} \ ,$$

that is,

$$\begin{aligned} S(t) &= \exp\left(\int_0^1 0\mathrm{d}u\right) = 1 \ , \\ \sigma(t)^2 &= \int_0^1 \frac{2\sigma_{\min}^2\sigma_{\mathrm{d}}^{2u}\log\sigma_{\mathrm{d}}}{1}\mathrm{d}u = \sigma_{\min}^2\left(\sigma_{\mathrm{d}}^{2t}-1\right) \ . \end{aligned}$$

B.3 Conditional Time Score

This section recaps and extends the analyses of Guth et al. (2025) and Yu et al. (2025), providing a unified presentation of the conditional time score framework. Using the stochastic process introduced in Equation (1), the log-density evolution can be written as

$$\begin{aligned} \partial_t \log p_t(y) &= \partial_t \log \int q_t(x)p_t(y|x)\mathrm{d}x \\ &= \frac{1}{p_t(y)}\left(\int p_t(y|x)\partial_t q_t(x) + q_t(x)\partial p_t(y|x)\mathrm{d}x\right) \\ &= \int \frac{q_t(x)p_t(y|x)}{p_t(y)}(\partial_t \log q_t(x) + \partial_t \log p_t(y|x))\mathrm{d}x \\ &= \mathbb{E}\left[\partial_t \log q_t(X_t) + \partial_t \log p_t(Y_t|X_t)|Y_t = y\right] \ , \end{aligned} \tag{30}$$

where

$$\begin{aligned} \partial_t \log p_t(y|x) &= \partial_t \log \mathcal{N}(y; x, \gamma(t)^2 I) \\ &= \partial_t\left(-\frac{d}{2}\log 2\pi - d\log\gamma(t) - \frac{\|y-x\|^2}{2\gamma(t)^2}\right) \\ &= -d\frac{\dot\gamma(t)}{\gamma(t)} + \frac{\|y-x\|^2}{\gamma(t)^2}\frac{\dot\gamma(t)}{\gamma(t)} \ , \end{aligned}$$

while $\partial_t \log q_t(X_t)$ doesn't have a general form and therefore should be treated case-by-case.

In this work, we mainly discuss special cases of Equation (1), where $(X_t)_{t\in[0,T]}$ is deterministic once the source(s), *i.e.* X_0 for DMs and (X_0, X_1) for SIs, are given. For simplicity, we define $\xi \sim \mu$ as the source and $X_t = T_t(\xi)$ is obtained by a deterministic map T_t (see examples bellow). In such case, the marginal of X_t can be defined by a Dirac delta

$$q_t(x) = \int \mu(\xi)\delta(x - T_t(\xi))\mathrm{d}\xi \Rightarrow p_t(y) = \int \mu(\xi)p_t(y|T_t(\xi))\mathrm{d}\xi \ .$$

Therefore, Equation (30) can be written as

$$\begin{aligned} \partial_t \log p_t(y) &= \partial_t \log \int \mu(\xi)p_t(y|T_t(\xi))\mathrm{d}\xi \\ &= \frac{1}{p_t(y)}\int \mu(\xi)p_t(y|T_t(\xi))\partial_t \log p_t(y|T_t(\xi))\mathrm{d}\xi \\ &= \int p(\xi|y)\partial_t \log \mathcal{N}(y; T_t(\xi), \gamma(t)^2 I)\mathrm{d}\xi \\ &= \mathbb{E}\left[-d\frac{\dot\gamma(t)}{\gamma(t)} - \frac{(Y_t - T_t(\xi))^\top \partial_t T_t(\xi)}{\gamma(t)^2} + \frac{\|Y_t - T_t(\xi)\|^2}{\gamma(t)^2}\frac{\dot\gamma(t)}{\gamma(t)}\Big|Y_t = y\right] \ . \end{aligned}$$

Example 1: Diffusion Models As introduced in Section 2.1, $(X_t)_{t\in[0,T]}$ in DMs are defined as $X_t = S(t)X_0$ with $X_0 \sim \pi$ and $S : \mathbb{R} \mapsto \mathbb{R}$. Therefore, $\xi = X_0$, $\mu = \pi$, and $T_t(\xi) = S(t)\xi$:

$$\partial_t \log p_t(y) = \mathbb{E}\left[-d\frac{\dot\gamma(t)}{\gamma(t)} - \frac{\dot S(t)X_0^\top(y - S(t)X_0)}{\gamma(t)^2} + \frac{\|y - S(t)X_0\|^2}{\gamma(t)^2}\frac{\dot\gamma(t)}{\gamma(t)}\right] . \tag{31}$$

Example 2: Stochastic Interpolants As introduced in Section 2.1, SIs are defined by two ends, *i.e.* $X_t = I_t(X_0, X_1)$ given (X_0, X_1). Therefore, $\xi = (X_0, X_1)$, μ any coupling of p_0, p_1 with marginals p_0, p_1, and $T_t(\xi_0, \xi_1) = I_t(\xi_0, \xi_1)$.

$$\partial_t \log p_t(y) = \mathbb{E}\left[-d\frac{\dot\gamma(t)}{\gamma(t)} - \frac{\partial_t I_t(X_0,X_1)^\top (y - I_t(X_0,X_1))}{\gamma(t)^2} + \frac{\|y - I_t(X_0,X_1)\|^2}{\gamma(t)^2}\frac{\dot\gamma(t)}{\gamma(t)}\right]. \tag{32}$$

B.4 Learning Log-densities with self-consistency

In this section, we will introduce two strategies to learn log-densities via enforcing their self-consistency relations: the Fokker-Planck regularization (Sun et al., 2024; Shi et al., 2024; Plainer et al., 2025) and the Bayes (or RNE) regularization He et al. (2025b). Such relations naturally arise from the underlying dynamics of the process and can be exploited to design training objectives. These methods typically rely on stronger assumptions about the generative process Y_t, which is why we primarily focus on the well-structured settings of DMs and SIs.

B.4.1 Consistency from Fokker–Planck Equation

Assume that the dynamic of the process $(Y_t)_t$ can be described by an (additive-noise) Itô SDE

$$dY_t = \alpha_t(Y_t)\mathrm{d}t + \beta_t \mathrm{d}W_t, \tag{33}$$

where $\alpha_t : [0,T] \times \mathbb{R}^d \to \mathbb{R}^d$ and $\beta_t : [0,T] \to \mathbb{R}_+$. Then the evolution of densities induced by this process is governed by the *Fokker-Planck Equation* (FPE), (see Appendix B.1 for more details) given by

$$\partial_t p_t + \nabla \cdot (\alpha_t p_t) - \frac{\beta_t^2}{2}\Delta p_t = 0\,, \tag{34}$$

where $\nabla\cdot$ is the divergence operator and Δ is the Laplacian operator both with respect to y. This formulation arises when the stochastic process admits an SDE representation. For example, in Diffusion Models we have $\alpha_t(y) = f(t)y$ and $\beta_t = g(t)$, while in Stochastic Interpolants $\alpha_t(y) = \mathbb{E}[\partial_t I_t(Y_0, Y_1) + \dot\gamma(t)Z | Y_t = y] + \frac{1}{2}\beta_t^2 \nabla \log p_t(y)$ and $\beta_t = g(t)$ [8]. Although the Fokker–Planck equation (34) is formulated in terms of the density p_t, it can equivalently be rewritten in terms of the log-density $\log p_t$ as

$$\partial_t \log p_t - \mathcal{F}_t(p_t) = 0, \quad \mathcal{F}_t(p_t) = \frac{\beta_t^2}{2}\left[\Delta \log p_t(y) + \|\nabla p_t(y)\|^2\right] - \alpha_t \cdot \nabla \log p_t - \nabla \cdot \alpha_t\,.$$

To enhance the accuracy of $(p_t^\theta)_t$, recent works (Sun et al., 2024; Shi et al., 2024; Plainer et al., 2025) propose enforcing its self-consistency by optimizing the following objective

$$\mathcal{L}_{\mathrm{FPE}}(\theta) = \mathbb{E}_t[\mathcal{L}_{\mathrm{FPE}}(\theta; t)], \quad \mathcal{L}_{\mathrm{FPE}}(\theta; t) = \mathbb{E}_{p_t}\left[\left(\partial_t \log p_t^\theta(Y_t) - \mathcal{F}_t(p_t^\theta)(Y_t)\right)^2\right]\,. \tag{35}$$

Although Shi et al. (2024) claim that this approach overcomes the blindness of score matching, we demonstrate in Appendix D that it remains susceptible to the same issue. The objective can also be combined with DSM, and a pretrained score estimator may be directly incorporated into $\mathcal{F}$ without further optimization. However, the method is computationally demanding, as training requires back-propagating through high-order derivatives of U^θ (specifically, the time derivative, the score, and the Laplacian) resulting in a substantial increase in cost. To mitigate this, Plainer et al. (2025) propose approximating the time-score using finite differences and estimating the residual term $\mathcal{F}_t$ with an unbiased estimator.

B.4.2 Consistency from Bayes Rule

A complementary perspective arises by considering the correlation between the distribution at two consecutive times $0 < s < t < T$. In this case, the marginal densities of Y_s and Y_t are connected through Bayes' rule

$$p_t(y_t) p_{s|t}(y_s|y_t) = p_s(y_s) p_{t|s}(y_t|y_s), \quad \text{for all } y_s, y_t \in \mathbb{R}^d\,, \tag{36}$$

[8] Note that $\alpha_t = v_t$ and $\beta_t = \sqrt{2\dot\gamma(t)\gamma(t)}$ is also a valid choice but we keep the previous decomposition to make the following proofs easier.

where $p_{t|s}$ and $p_{s|t}$ denote the conditional distributions of Y_t given Y_s and Y_s given Y_t, respectively. Although these conditional distributions are generally intractable, they admit tractable approximations when $(Y_t)_t$ is defined as the solution of an SDE, as in DMs or SIs. We should first note that the time-reversal, which generally exists for DMs or SIs, of Equation (33) is written as

$$\mathrm{d}Y_t = [\alpha_t(Y_t) - \beta_t^2 \nabla \log p_t(Y_t)]\mathrm{d}t + \beta_t \mathrm{d}\tilde{W}_t \ , \tag{37}$$

where $(\tilde{W}_t)_t$ is a standard Brownian motion in reversed time. By letting $\delta = t - s$, one can approximate the transition kernels via an Euler–Maruyama (EM) discretization of the corresponding SDEs (8) and (9), yielding

$$p_{t|s}(\cdot|y_s) \approx \mathcal{N}\left(y_s + \delta\alpha_s(y_s),\ \delta\beta_s^2 \mathrm{I}_d\right) \ , \tag{38}$$

$$p_{s|t}(\cdot|y_t) \approx \mathcal{N}\left(y_t - \delta\left[\alpha_t(y_t) - \beta_t^2 \nabla \log p_t(y_t)\right],\ \delta\beta_t^2 \mathrm{I}_d\right) \ . \tag{39}$$

In He et al. (2025b), the authors propose to regularize the sequence $(\log p_t^\theta)_t$ by minimizing the squared discrepancy between the logarithm of the two sides of the Bayes rule (36)

$$\mathcal{L}_{\mathrm{Bayes}}(\theta) = \mathbb{E}_{s,t}[\mathcal{L}_{\mathrm{Bayes}}(\theta; s, t)] \ ,$$

with

$$\mathcal{L}_{\mathrm{Bayes}}(\theta; s, t) = \mathbb{E}_{p_s, p_t}\left[\left(\log p_s^\theta(Y_s) - \log p_t^\theta(Y_t) + \log p_{t|s}^\theta(Y_t|Y_s) - \log p_{s|t}^\theta(Y_s|Y_t)\right)^2\right] \ ,$$

where $p_{t|s}^\theta$ and $p_{s|t}^\theta$ denote approximations of $p_{t|s}$ (38) and $p_{s|t}$ (39), obtained by replacing $\nabla \log p_t$ with its approximation $\nabla \log p_t^\theta$. As highlighted in He et al. (2025b), for DMs, the specific choices of f and g allow a closed-form expression for the forward-time kernel $p_{t|s}$. This yields a more accurate approximation of the time-forward kernel than the EM scheme (38) and avoids reliance on intractable quantities such as the score. However, even in this favorable case, the time-backward kernel (39) remains approximate, introducing bias that breaks self-consistency. Exact self-consistency is only recovered in the small-step limit ($\delta \to 0$), where the EM scheme becomes accurate.

B.4.3 Bayes and Fokker-Plank regularizations: Discrete v.s. Continuous

Now, we are going to show that $\mathcal{L}_{\mathrm{Bayes}}$ and $\mathcal{L}_{\mathrm{FPE}}$ are asymptotically related as follows:

Proposition 6. *Let $\delta > 0$. In the small step-size regime, the Bayes objective $\mathcal{L}_{\mathrm{Bayes}}$ recovers the Fokker–Planck regularization $\mathcal{L}_{\mathrm{FPE}}$, i.e.,*

$$\lim_{\delta \to 0} \frac{1}{\delta} \mathcal{L}_{\mathrm{Bayes}}(\theta; t, t+\delta) = \mathcal{L}_{\mathrm{FPE}}(\theta; t) \ . \tag{40}$$

The proof is provided at appendix E.5. This result is also closely related to the derivations in (Skreta et al., 2025b, Appendix D.2). Building on this observation, Proposition 6 implies that $\mathcal{L}_{\mathrm{Bayes}}$ is either in the small step-size regime, where it remains mode blind, or in the non-small step-size regime, where it becomes biased.

C Additional connection to related works

C.1 Connection to related works in training energy-based generative models

Figure 6 illustrates the connections between `DiffCLF` and other existing methods that train energy-based generative models (usually diffusion models), while Table 2 summurizes the computational overhead, assumptions/requirements, and mode-blindness issue for those methods.

DRE-∞ (Choi et al., 2022), C*t*SM (Yu et al., 2025; Guth et al., 2025), and FPE-reg (Plainer et al., 2025) train models such that their time derivatives match the ground-truth ones, which is termed Time Score Matching (*t*SM) (Choi et al., 2022). Jointly training with DSM, the optimality yields $\nabla \log p_t^\theta = \nabla \log p_t$ and $\partial_t \log p_t^\theta = \partial_t \log p_t$. However, it is not sufficient to reach the optimality of log density, i.e. $\log p_t^\theta = \log p_t$, especially when the modes are disconnected (see Appendix D for more discussions).

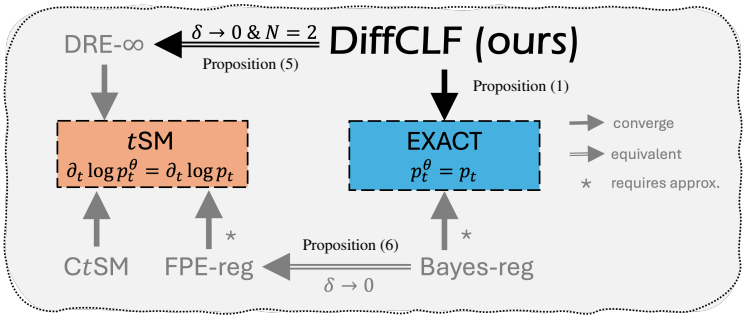


Figure 6: Connections between `DiffCLF` (ours) and other related works.

Table 2: Comparison of properties between `DiffCLF` (ours) and related methods. For *Number of Function Evaluation* (NFE), we report the actual evaluations when jointly training with $\mathcal{L}_{\text{DSM}}$, where we treat auto-differentiation costing roughly twice a network forward pass. For DRE-∞, though Choi et al. (2022) proposes to directly parameterize the score, we choose to treat it a way for energy-based training and therefore report the NFE for energy-parameterization, which doubles the original calculation. For FPE-reg, we follow the approximations made in Plainer et al. (2025) and count their NFE.

Method	NFE	No approx.	Prior knowledge of $(p_t)_t$	mode-weight aware
DRE-∞ (Choi et al., 2022)	4	✗	–	✗
CtSM (Yu et al., 2025; Guth et al., 2025)	2	✓	$\partial_t X_t$	✗
FPE-reg (Plainer et al., 2025)	14	✗	Induced from known FPE	✗
Bayes-reg (He et al., 2025b)	3	✗	$p_{t\|s}$ & $p_{s\|t}$	✗
N-`DiffCLF` (ours)	$N+1$	✓	–	✓

Bayes-reg (He et al., 2025b) leverages the transition kernels for energy-based training, which is guaranteed to reach optimal $(p_t^\theta)_t$ if training with arbitrary time pairs (t, t'). However, in practical cases such as DMs or SIs, the transition kernels can only be approximated when t and t' close enough, through a Euler–Maruyama discretization of the dynamic's SDE. Proposition 6 shows that when t and t' are close enough, Bayes-reg recovers FPE-reg, and therefore, it shares the same issues in tSM.

`DiffCLF` treats the energy training as classification tasks, which, in the 2-class case, recovers tSM in the continuous-time limit (see Proposition 5). In fact, when (t, t') in Equation (12) are close enough, one could approximate

$$\log p_{t'}^\theta(x) = \log p_t^\theta(x) + \partial_t \log p_t^\theta(x)(t - t') + \mathcal{O}(|t - t'|^2) \, .$$

Therefore, the first term in Equation (12) can be approximated as [9]

$$\mathbb{E}_{p_t}\left[\log \frac{p_t^\theta(X)}{p_t^\theta(X) + p_{t'}^\theta(X)}\right] = \mathbb{E}_{p_t}\left[\log \frac{1}{1 + \exp(\partial_t \log p_t^\theta(X)(t - t') + \mathcal{O}(|t - t'|^2))}\right] , \tag{41}$$

$$\approx \mathbb{E}_{p_t}\left[\log\left(\frac{1}{2} + \frac{1}{4}\partial_t \log p_t^\theta(X)(t - t')\right)\right] . \tag{42}$$

Analogously, the second term in Equation (12) can be approximated as

$$\mathbb{E}_{p_{t'}}\left[\log \frac{p_{t'}^\theta(X)}{p_t^\theta(X) + p_{t'}^\theta(X)}\right] \approx \mathbb{E}_{p_{t'}}\left[\log\left(\frac{1}{2} - \frac{1}{4}\partial_t \log p_t^\theta(X)(t - t')\right)\right] . \tag{43}$$

[9] This is induced by the first order approximation of the log sigmoid function, i.e. $\log \frac{1}{1+e^{-z}} = -\log 2 + \frac{z}{2} + \mathcal{O}(z^2)$.

DRE-∞ (Choi et al., 2022) proposes to parameterize a time-score network s_θ and optimize Equation (12) with approximations given by Equations (42) and (43):

$$\mathcal{L}_{\mathrm{DRE}-\infty}(\theta) = \mathbb{E}_t[\mathcal{L}_{\mathrm{DRE}-\infty}(\theta; t, \delta)] \tag{44}$$

$$\text{with } \mathcal{L}_{\mathrm{DRE}-\infty}(\theta; t, \delta) = -\frac{1}{2}\mathbb{E}_{p_t}[\log(1 - h_\theta(Y, t, \delta))] - \frac{1}{2}\mathbb{E}_{p_{t+\delta}}[\log h_\theta(Y, t, \delta)] \,, \tag{45}$$

$$h_\theta(Y, t, \delta) = \frac{1}{2} + \frac{1}{4} s_\theta(x, t)\delta \,. \tag{46}$$

C.2 Connection to MBAR

Computing free-energy differences, differences in log-normalizing constants between two potentials, is a central task in statistical physics and molecular dynamics (Lelièvre et al., 2010). In this section, we first introduce the free-energy estimation problem and the golden standard method MBAR (Shirts & Chodera, 2008) and build connection between with `DiffCLF`, which shows that their difference is the choice of model-parameterization.

Free-energy estimation and FEP. Given a distribution

$$p(y) = \frac{1}{\mathcal{Z}} \exp(-\mathrm{U}(y)), \quad \text{with} \quad \mathcal{Z} = \int_\Omega \exp(-\mathrm{U}(y))\mathrm{d}y \,,$$

where $\Omega \subseteq \mathbb{R}^d$ is the support and $\mathrm{U} : \Omega \to \mathbb{R}$ is the energy function, the *free energy* is defined as the negative log-partition function, i.e.

$$\mathrm{F} = -\log \mathcal{Z} = \int_\Omega \exp(-U(y))\mathrm{d}y \,.$$

While direct estimation of F is difficult, one typically estimates the *free-energy difference* between two states A and B with supports Ω_A, Ω_B and energy functions $\mathrm{U}_A, \mathrm{U}_B$ respectively,

$$\Delta \mathrm{F}_{AB} = \mathrm{F}_A - \mathrm{F}_B = \log \frac{Z_B}{Z_A}.$$

The easiest way to estimate ΔF_{AB} is through Free Energy Perturbation (FEP Zwanzig, 1954) , which reformulates the free energy difference estimation task as an importance sampling problem:

$$\Delta F_{AB} = \log \mathbb{E}_{p_A}[\exp(U_A(x) - U_B(x))] = -\log \mathbb{E}_{p_B}[\exp(U_B(x) - U_A(x))] \,. \tag{47}$$

MBAR. The golden-standard method for estimating this free-energy difference is *Multi-state Bennett Acceptance Ratio* (MBAR) (Shirts & Chodera, 2008), which generalizes *Bennett Acceptance Ratio* (BAR) (Bennett, 1976) to multi-states by introducing (i) a sequence of intermediate distributions with tractable energy functions bridging the two states $\{\mathrm{U}_i\}_{n=1}^N$ and (ii) associated samples with these distributions $\{y_n^{(k)}\}_{n=1,k=1}^{N,K_n}$ such that

$$\mathrm{U}_0 = \mathrm{U}_A, \quad \mathrm{U}_N = \mathrm{U}_B, \quad p_n(y) = \frac{\exp(-\mathrm{U}_n(y))}{\mathcal{Z}_n}, \quad y_n^k \sim p_n \,,$$

where $Z = \mathcal{Z}_n = \int_{\Omega_n} \exp(-\mathrm{U}_n(y))\mathrm{d}y$. Let $\mathrm{F}_n = -\log \mathcal{Z}_n$, by defining

$$p(c = n|y) = \frac{\exp(-\mathrm{U}_n(y) + \mathrm{F}_n)}{\sum_{m=1}^N \exp(-\mathrm{U}_m(y) + \mathrm{F}_m)} \quad \text{and} \quad p(y|n) = p_n(y) \,.$$

MBAR treats the free energy estimation problem as a multiclass classification problem with maximum likelihood

$$\begin{aligned}
\mathrm{F}_{1:N}^* &= \arg\max_{\mathrm{F}_{1:N}} \mathbb{E}_{p(n)}\mathbb{E}_{p_n}[\log p(c = n|y)] \\
&= \arg\max_{\mathrm{F}_{1:N}} \mathbb{E}_{p(n)}\mathbb{E}_{p_n}\left[\frac{\exp(-\mathrm{U}_n(y) + \mathrm{F}_n)}{\sum_{m=1}^N \exp(-\mathrm{U}_m(y) + \mathrm{F}_m)}\right] \\
&\approx \arg\max_{\mathrm{F}_{1:N}} \frac{1}{N}\sum_{n=1}^N \frac{1}{K_n}\sum_{k=1}^{K_n} \log \frac{\exp(-\mathrm{U}_n(y_n^{(k)}) + \mathrm{F}_n)}{\sum_{m=1}^N \exp(-\mathrm{U}_m(y_n^{(k)}) + \mathrm{F}_m)} \,,
\end{aligned} \tag{48}$$

which solved using a fixed point iteration (see (Shirts & Chodera, 2008, Equation 3)) or using the Newton-Raphson algorithm (see (Shirts & Chodera, 2008, Equation 6)). Although the optimization problem is easy to solve, it requires equilibrium samples $\{y_n^{(k)}\}_k$ from each intermediate distribution p_n, where the intermediate energy functions are usually defined by tempering e.g.

$$\mathrm{U}_n(y) = (1-\beta_n)\mathrm{U}_A(y) + \beta_n \mathrm{U}_B(y) \quad \text{, with } \beta_1 = 0 \text{ and } \beta_N = 1 \ . \tag{49}$$

To get equilibrium samples from each intermediate distributions, one typically used annealed Markov Chain Monte Carlo samplers which are expensive.

Connection between MBAR and `DiffCLF`. By comparison with the `DiffCLF` objective (48) and the MBAR objective (10), one could observe that the difference between MBAR and `DiffCLF` is the *model-parameterization*.

- In MBAR, the energy functions $(\mathrm{U}_t)_t$ [10] are assumed known and the learnable objects are only the free energies $(\mathrm{F}_t)_t$, i.e.

$$p_t^\theta(y) = \exp(-U_t(y) + \mathrm{F}_t^\theta) \ .$$

- In `DiffCLF`, the EBMs are fully parameterized (see Equation (2)).

Besides, the accuracy of MBAR critically depends on the choice of the path, e.g. the anealling temperatures. Similarly to the BG case, learned energies could directly provides a data-driven approach to construct more effective paths, potentially surpassing hand-crafted designs.

D Blindness of Score Matching and Time Score Matching

In this section, we revisit the blindness of score matching, first analyzed in Wenliang & Kanagawa (2021); Zhang et al. (2022), and show that the problem persists even when matching higher-order derivatives or the time derivative of the trajectory $(\log p_t)_t$.

Let a set of time-dependent distributions with differentiable densities $\{g_t^1, \dots, g_t^K\}$ having mutual disjoint (disconnected) support sets $\{\mathcal{X}_t^1, \dots, \mathcal{X}_t^K\}$, where $\mathcal{X}_t^i \cap \mathcal{X}_t^j = \emptyset$ for any $i \neq j$ [11] For all $t \in [0, T]$, we define two mixture distributions $p_t = \sum_{k=1}^K \alpha_t^k g_t^k$ and $q_t = \sum_{k=1}^K \beta_t^k g_t^k$, where $\sum_{k=1}^K \alpha_t^k = 1$ and $\sum_{k=1}^K \beta_t^k = 1$.

Score Matching is blind. Score Matching is minimizing the *Fisher Divergence* (FD) between p_t and q_t. Let $t \in [0, T]$, following the (Zhang et al., 2022, Proposition 1) we have that

$$\begin{aligned}
&\mathrm{FD}(p_t, q_t) := \mathbb{E}_{p_t}\left[\|\nabla \log p_t(X_t) - \nabla \log q_t(X_t)\|\,|^2\right] \ , \\
&= \sum_{k=1}^K \int_{\mathcal{X}_t^k} \left\| \frac{\nabla(\sum_{j=1}^K \alpha_t^j g_t^j(x))}{p_t(x)} - \frac{\nabla(\sum_{j=1}^K \beta_t^j g_t^j(x))}{q_t(x)} \right\|^2 \alpha_t^k g_t^k(x) \mathrm{d}x \ , \\
&= \sum_{k=1}^K \int_{\mathcal{X}_t^k} \left\| \frac{\cancel{\alpha_t^k} \nabla g_t^k(x)}{\cancel{\alpha_t^k} g_t^k(x)} - \frac{\cancel{\beta_t^k} \nabla g_t^k(x)}{\cancel{\beta_t^k} g_t^k(x)} \right\|^2 \alpha_t^k g_t^k(x) \mathrm{d}x = 0,
\end{aligned}$$

using $g_t^i(x_j) = 0$ and $\nabla g_t^i(x_j) = 0$ for $\forall x_j \in \mathcal{X}_j$ when $i \neq j$. Therefore, SM (or FD) is ill-defined on disconnected sets and has the blindness problem.

Remark 7. *The marginal p_t needs not be supported on disjoint sets for all $t \in [0, T]$. For example, in DMs the terminal distribution p_T is typically Gaussian and thus fully connected. In such cases, mixture proportions may be correctly estimated. However, at intermediate times where the support is disconnected, the proportions can be misestimated. Since the objective averages losses across time, the blindness issue may persist overall.*

[10] For clarity, we change the index of n to a set of discretized time t as in `DiffCLF`.

[11] Note that multi-modality could be modified depending on t by simply setting two components and respective weights equal.

Higher-order Score Matching is blind. As suggested in Lu et al. (2022), one could minimize the divergence between the Hessian or Laplacian of log-densities, i.e.

$$\mathbb{E}_{p_t}\left[\left\|\nabla^2 \log p_t(X_t) - \nabla^2 \log q_t(X_t)\right\|_F^2\right] \quad \text{or} \quad \mathbb{E}_{p_t}\left[\left(\Delta \log p_t(X_t) - \Delta \log q_t(X_t)\right)^2\right] ,$$

where $\|\cdot\|_F$ is the Frobenius norm. Similarly as the previous paragraph, for all $t \in [0, T]$ and $x \in \mathcal{X}_t^k$,

$$\begin{aligned}\nabla^2 \log p_t(x) &= \frac{p_t(x)\sum_{j=1}^K \alpha_t^j \nabla^2 g_t^j(x) - \left(\sum_{j=1}^K \alpha_t^j \nabla g_t^j(x)\right)\nabla p_t(x)^\top}{p_t(x)^2} , \\ &= \frac{\cancel{\alpha_t^k} g_t^k(x) \cancel{\alpha_t^k} \nabla^2 g_t^k(x) - \cancel{\alpha_t^k} \nabla g_t^k(x) \cancel{\alpha_t^k} \nabla g_t^k(x)^\top}{(\cancel{\alpha_t^k} g_t^k(x))^2} ,\end{aligned}$$

which doesn't depend on the weights α. The cases for Laplacian or other higher-order (w.r.t. x) regression are analogous.

Time Score Matching can be blind. The Time Score Matching (see Appendix B.3 for details) objective can be written for any $t \in [0, T]$ as

$$\begin{aligned}&\mathbb{E}_{p_t}\left[\left(\partial_t \log p_t(X_t) - \partial_t \log q_t(X_t)\right)^2\right] = \\ &\qquad \sum_{k=1}^K \int_{\mathcal{X}_t^k} \left(\frac{\partial_t(\sum_{j=1}^K \alpha_t^j g_t^j(x))}{p_t(x)} - \frac{\partial_t(\sum_{j=1}^K \beta_t^j g_t^j(x))}{q_t(x)}\right)^2 \alpha_t^k g_t^k(x)\mathrm{d}x , \quad (50)\end{aligned}$$

where $\partial_t(\sum_{j=1}^K \alpha_t^j g_t^j(x))$ can be expanded by leveraging $g_t^i(x_j) = 0$ for any $x_j \in \mathcal{X}_j$ and $i \neq j$

$$\partial_t\left(\sum_{j=1}^K \alpha_t^j g_t^j(x)\right) = (\partial_t \alpha_t^k) g_t^k(x) + \sum_{j=1}^K \alpha_t^j \partial_t g_t^j(x), \quad \forall x \in \mathcal{X}_t^k , \quad (51)$$

hence,

$$\begin{aligned}&\mathbb{E}_{p_t}\left[\left(\partial_t \log p_t(X_t) - \partial_t \log q_t(X_t)\right)^2\right] \\ &= \sum_{k=1}^K \int_{\mathcal{X}_t^k} \left(\frac{(\partial_t \alpha_t^k) g_t^k(x) + \sum_{j=1}^K \alpha_t^j \partial_t g_t^j(x)}{\alpha_t^k g_t^k(x)} - \frac{(\partial_t \beta_t^k) g_t^k(x) + \sum_{j=1}^K \beta_t^j \partial_t g_t^j(x)}{\beta_t^k g_t^k(x)}\right)^2 \alpha_t^k g_t^k(x)\mathrm{d}x , \\ &= \sum_{k=1}^K \int_{\mathcal{X}_t^k} \left(\frac{\partial_t \alpha_t^k}{\alpha_t^k} - \frac{\partial_t \beta_t^k}{\beta_t^k} + \frac{\sum_{j\neq k} \alpha_t^j \partial_t g_t^j(x)}{\alpha_t^k g_t^k(x)} - \frac{\sum_{j\neq k} \beta_t^j \partial_t g_t^j(x)}{\beta_t^k g_t^k(x)}\right)^2 \alpha_t^k g_t^k(x)\mathrm{d}x \geq 0 ,\end{aligned}$$

which can be 0 in some cases. For example, if the mixture weights α, β are time-independent (as in DMs) and the supports of g_t vary only slowly over time (also typical in DMs), then the loss becomes mode-blind.

The Fokker-Planck regularization is mode blind. When p_t, q_t are generated from the same Itô SDE, say $dX_t = f(t, X_t)\mathrm{d}t + g(t, X_t)\mathrm{d}W_t$, starting from p_0, q_0 respectively, the Fokker-Planck Equation (see Appendix B.1) tells us

$$\begin{aligned}\partial_t \log p_t(x) &= -\nabla \cdot f(t,x) - f(t,x)^\top \nabla \log p_t(x) + \frac{1}{2} g(t,x)^\top g(t,x)\left(\Delta \log p_t(x) + \|\nabla \log p_t\|^2\right) , \\ \partial_t \log q_t(x) &= -\nabla \cdot f(t,x) - f(t,x)^\top \nabla \log q_t(x) + \frac{1}{2} g(t,x)^\top g(t,x)\left(\Delta \log q_t(x) + \|\nabla \log q_t\|^2\right) .\end{aligned}$$

Therefore, the FPE regularization is equivalent to regressing a combination of (i) the score and its squared norm, and (ii) the Laplacian, which are all mode-blind. Therefore, FPE regularization in such cases is blind in this case.

Alleviating the blindness of score matching. Several approaches have been proposed to address mode blindness. Zhang et al. (2022) introduce an auxiliary noise distribution m and minimize the Fisher divergence between mixtures of (π, m) and (p^θ, m), which indirectly enforces proximity between π and p^θ. Similarly, Schröder et al. (2023) propose the energy discrepancy, a contrastive objective between π and its noisy counterpart that provably avoids blindness (see (Schröder et al., 2023, Figure 1)). However, both methods hinge on the careful design and tuning of the auxiliary noise distribution or kernel, limiting their scalability in practice.

E Proofs of propositions

E.1 Proof of proposition 1

Proposition 1. *Let $N \in \mathbb{N}^*$ and $t_{1:N} \in [0,T]^N$. The Diffusive Classification loss is*

$$\mathcal{L}_{\mathrm{clf}}(\theta; N) = \mathbb{E}_{t_{1:N}}[\mathcal{L}_{\mathrm{clf}}(\theta; t_{1:N})], \quad \mathcal{L}_{\mathrm{clf}}(\theta; t_{1:N}) = -\frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{p_{t_i}}\left[\log \frac{p^\theta_{t_i}(Y_i)}{\sum_{j=1}^N p^\theta_{t_j}(Y_i)}\right] ,$$

where $p^\theta_t(y_t) = \exp(-\mathrm{U}^\theta_t(y_t) + \mathrm{F}^\theta_t)$. A minimizer of $\mathcal{L}_{\mathrm{clf}}(\theta)$ is attained by $p^{\theta^*}_t = p_t$ for all $t \in [0,T]$.

Proof. We first show that, given a batch of times $t_{1:N} \in [0,T]^N$, $p^{\theta *}_{t_i} = p_{t_i}$ for any $i \in [\![1, N]\!]$ obtains the optimality. For clarity, for all $i \in [\![1, N]\!]$, set $p_i = p_{t_i}$ and $p^\theta_i = p^\theta_{t_i}$. Let $\Omega_i = \mathrm{supp}(p_i)$ and $\Omega = \bigcup_{i=1}^N \Omega_i$. Extend each p_i by 0 on $\Omega \setminus \Omega_i$ so that all integrals are over Ω. Then

$$\sum_{i=1}^N \mathbb{E}_{p_i}\left[\log \frac{p^\theta_i(Y_{t_i})}{\sum_{j=1}^N p^\theta_j(Y_{t_i})}\right] = \int_\Omega \sum_{i=1}^N p_i(y) \log \frac{p^\theta_i(y)}{\sum_{j=1}^N p^\theta_j(y)} \,\mathrm{d}y \,. \tag{52}$$

For $y \in \Omega$, define

$$s(y) := \sum_{k=1}^N p_k(y), \qquad w_i(y) := \frac{p_i(y)}{s(y)}, \qquad w^\theta_i(y) := \frac{p^\theta_i(y)}{\sum_{k=1}^N p^\theta_k(y)} \,,$$

interpreting the ratios arbitrarily if $s(y) = 0$ (those y do not affect the integral). Then the integrand rewrites as

$$\sum_{i=1}^N p_i(y) \log w^\theta_i(y) \;=\; s(y) \sum_{i=1}^N w_i(y) \log w^\theta_i(y) \,.$$

Since $\sum_i w_i(y) = \sum_i w^\theta_i(y) = 1$, Gibbs' inequality gives, for all y with $s(y) > 0$,

$$\sum_{i=1}^N w_i(y) \log w^\theta_i(y) \;\le\; \sum_{i=1}^N w_i(y) \log w_i(y) \,,$$

with equality iff $w^\theta_i(y) = w_i(y)$ for all i. Multiplying by $s(y)$ and integrating over Ω yields

$$\sum_{i=1}^N \mathbb{E}_{p_i}\left[\log \frac{p^\theta_i(Y_i)}{\sum_{j=1}^N p^\theta_j(Y_i)}\right] \le \int_\Omega \sum_{i=1}^N p_i(y) \log \frac{p_i(y)}{\sum_{j=1}^N p_j(y)} \,\mathrm{d}y \,,$$

where equality holds whenever

$$\frac{p^\theta_i(y)}{\sum_{j=1}^N p^\theta_j(y)} = \frac{p_i(y)}{\sum_{j=1}^N p_j(y)} \quad \text{for } \bar{p}\text{-a.e. } y \in \Omega \,,$$

with $\bar{p} = \frac{1}{N}\sum_i p_i$. In particular, choosing $p^\theta_i = p_i$ for all i satisfies the condition and attains the minimum. Therefore, choosing $p^\theta_t = p_t$ for all $t \in [0,T]$ attains the minimum of $\mathcal{L}_{\mathrm{clf}}(\theta)$. □

E.2 PROOF OF PROPOSITION 2

It is noticeable that optimizing the `DiffCLF` objective solely can achieve different minimums. For example, one could easily verify that $p_t^\theta(x) = c(x)p_t(x)$ for a non-negative function c is also a minimum. In the following proposition, we will show that jointly optimizing with the *Denoising Score Matching* (DSM) objective (7) can fix this non-uniqueness issue.

To prove, we will first present the properties of minimizers for $\mathcal{L}_{\text{clf}}$ and $\mathcal{L}_{\text{DSM}}$ respectively, and then combine them to finish the proof.

Proposition 2 (Formal)**.** *Let $t \in [0, T]$, if conditions (A1) and (A2) are fulfilled:*

- *(A1) For any t, p_t and p_t^θ are absolutely continuous w.r.t. the Lebesgue measure on $\Omega_t := \text{supp}(p_t)$, with $\nabla \log p_t, \nabla \log p_t^\theta \in L^2(p_t)$,*
- *(A2) The union of supports $\Omega := \bigcup_{t\in[0,T]} \Omega_t$ is (path-)connected,*

then $p_t^{\theta^\star} = p_t$ for all $t \in [0, T]$ is the unique minimizer of the joint objective $\mathcal{L}_{\text{DSM}} + \mathcal{L}_{\text{clf}}$ (Equations (7) and (10)).

Proof. We first show that, given any batch of times $t_{1:N}$, by optimizing $\sum_{i=1}^N \mathcal{L}_{\text{DSM}}(\theta; t_i) + \mathcal{L}_{\text{clf}}(\theta; t_{1:N})$, then $p_{t_i}^\theta = p_{t_i}$ for all $i \in [\![1, N]\!]$ is the unique minimizer of the optimization problem. For clarity, let $p_i = p_{t_i}$ and $p_i^\theta = p_{t_i}^\theta$.

Step 1 (structure of $\mathcal{L}_{\text{clf}}$ minimizers). By proposition 1, any minimizer of $\mathcal{L}_{\text{clf}}(\theta; t_{1:N})$ satisfies, for $\bar{p}$-a.e. $y \in \Omega$ and any $i \in [\![1, N]\!]$,

$$\frac{p_i^\theta(y)}{\sum_{j=1}^N p_j^\theta(y)} = \frac{p_i(y)}{\sum_{j=1}^N p_j(y)} ,$$

where $\bar{p} := \frac{1}{N}\sum_{i=1}^N p_i$. Equivalently, there exists a positive measurable function $c : \Omega \to (0, \infty)$, independent of i (i.e. time-independent), such that

$$p_i^\theta(y) = c(y)\, p_i(y) \quad \text{for all } i,\ \bar{p}\text{-a.e. } y \in \Omega , \tag{53}$$

together with the per-class normalization constraints

$$\int_\Omega c(y)\, p_i(y)\, \mathrm{d}y = 1 \quad \text{for each } i = 1, \dots, N . \tag{54}$$

Step 2 (structure of $\mathcal{L}_{\text{DSM}}$ minimizers). By standard DSM identifiability (under (A1)), any minimizer of $\sum_{i=1}^N \mathcal{L}_{\text{DSM}}(\theta; t_i)$ satisfies

$$\nabla \log p_i^\theta(y) = \nabla \log p_i(y) \quad \text{for } p_i\text{-a.e. } y \in \Omega_i,\ \forall i \in [\![1, N]\!] . \tag{55}$$

Using Equation (53) in Equation (55) gives, for each i,

$$\nabla \log \big(c(y)\, p_i(y)\big) = \nabla \log p_i(y) \quad \Rightarrow \quad \nabla \log c(y) = 0 \quad \text{for } p_i\text{-a.e. } y \in \Omega_i .$$

Therefore, $c(y) \equiv C$ is constant $\bar{p}$-a.e. on Ω. Recalling the normalization constraint where $\int c(y)p_i(y)dy = 1$, we have $C = 1$ and hence $p_i^\theta = p_i$ a.e. for all i. Since the normalization function $c(y) \equiv 1$ is unique, $p_i^\theta = p_i$ is the unique minimizer. Hence, the joint objective $\mathcal{L}_{\text{DSM}}(\theta) + \mathcal{L}_{\text{clf}}(\theta)$ has an unique minimum, that is $p_t^\theta = p_t$ for all $t \in [0, T]$. □

E.3 PROOF OF PROPOSITION 3

In this part, we deliver the properties of estimator for the joint objective $\mathcal{L}_{\text{DSM}} + \mathcal{L}_{\text{clf}}$ (Equations (7) and (10)). We first consider the joint loss given a fixed set of times, $t_{1:N} \in [0, T]^N$. Then further extend it with consideration of the randomness of times to finish the proof.

E.3.1 LEMMATA

For clarity, let $q_{t_i} = q_i$, $p_{t_i} = p_i$, $p^\theta_{t_i} = p^\theta_i$, and $\mathrm{supp}(p_i) = \Omega_i$ for all $i \in [\![1, N]\!]$, and define

$$\mathcal{L}^M_{t_{1:N}}(\theta) = \frac{1}{M}\sum_{m=1}^{M} l_{t_{1:N}}(\theta; m), \quad l_{t_{1:N}}(\theta; m) = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{p^\theta_i(Y^{(m)}_i)}{\sum_{j=1}^N p^\theta_j(Y^{(m)}_i)} ,$$

$$\mathcal{J}^M_{t_{1:N}}(\theta) = \frac{1}{NM}\sum_{m=1}^{M}\sum_{i=1}^{N} j_{t_i}(\theta; m), \quad j_{t_i}(\theta; m) = \left\| \nabla \log p^\theta_i(\tilde{Y}^{(m)}_i) - \nabla \log p_i(\tilde{Y}^{(m)}_i | \tilde{X}^{(m)}_i) \right\|^2 ,$$

where

- $\mathcal{L}^M_{t_{1:N}}$ is the empirical `DiffCLF` loss;
- $\mathcal{J}^M_{t_{1:N}}$ is the empirical DSM loss;
- $Y^{(m)}_i \overset{i.i.d.}{\sim} p_i$ for all $i \in [\![1, N]\!]$;
- $\tilde{X}^{(m)}_i \overset{i.i.d.}{\sim} q_i$ for all $i \in [\![1, N]\!]$;
- $\tilde{Y}^{(m)}_i \sim p_i(\cdot|\tilde{X}^{(m)}_i)$ for all $i \in [\![1, N]\!]$;
- $Y^{(m)}_i, Y^{(m)}_i, \tilde{Y}^{(m)}_i$ are independent for all $i \in [\![1, N]\!]$.

Let $\Omega = \cup_{i=1}^N \Omega_i$ as the extended support, where $p_i(y) = 0$ for any $x \in \Omega/\Omega_i$ for all i.

Recall from Proposition 1, when jointly optimizing the DSM and `DiffCLF` losses, the optimal parameter $\theta^\star$ is unique and recovers the true marginal densities $(p_t)_t$ (or $(p_i)_i$ in the fixed time-batch setting). Therefore in the following, we have $p^{\theta^\star}_t = p_t$.

Let $\hat{\theta}^M_{t_{1:N}}$ optimizing the joint empirical loss $\mathcal{L}^M_{t_{1:N}} + \mathcal{J}^M_{t_{1:N}}$ and $\theta^\star_{t_{1:N}}$ optimizing the joint population loss $\mathcal{L}_{t_{1:N}} + \mathcal{J}_{t_{1:N}}$. For simplicity, we use $\theta^\star$ and $\theta^*_{t_{1:N}}$ interchangeably whenever no ambiguity arises. By leveraging the uniqueness of the joint optimum, we proceed in three steps:

(1) We generalize Lemmas 12–14 of Gutmann & Hyvärinen (2010) from the binary case to the multi-class setting with a fully parameterized model, obtaining Lemmas 11 to 13.

(2) We impose standard regularity conditions on the DSM objective. In particular, Assumptions 8 and 9 are classical assumptions in L_2-risk minimization and M-estimation, ensuring convergence in probability of the empirical Hessian and vanishing population gradient at $\theta^\star$. We additionally posit Assumption 10, which provides a finite-sample covariance structure for the DSM gradient. We do not attempt to verify Assumption 10 for specific architectures, as our main focus is the diffusive classification objective.

(3) Finally, combining Assumptions 8 to 10 with Lemmas 11 to 13, we show that $\hat{\theta}_{t_{1:N}}$ is a consistent estimator of $\theta^\star$, and we obtain its asymptotic covariance matrix together with an $\mathcal{O}(1/M)$ error bound around $\theta^\star$, in Lemma 14.

Assumption 8.

(i) $\nabla^2_\theta \left[M^{-1} \sum_m j_t(\theta; m) \Big|_{\theta=\theta^\star} \right]$ *converges in probability to a positive-definite matrix* $\mathcal{P}_t$ *as the sample size* M *tends to infinity.*

(ii) $\nabla^2_\theta \mathcal{J}^M_{t_{1:N}}$ *converges in probability to* $\mathcal{P}_{t_{1:N}} := N^{-1}\sum_{i=1}^N \mathcal{P}_{t_i}$ *as the sample size* M *tends to infinity. By (i),* $\mathcal{P}_{t_{1:N}}$ *is a positive-definite matrix.*

Assumption 9. $\mathbb{E}\left[\nabla_\theta \left(M^{-1} \sum_m j_t(\theta; m) \right) \Big|_{\theta=\theta^\star} \right] = 0$ *and* $\mathbb{E}\left[\nabla_\theta \mathcal{J}^M_{t_{1:N}}(\theta) \Big|_{\theta=\theta^\star} \right] = 0.$

Assumption 10.

$$\mathrm{Cov}\left[\nabla_\theta \left(M^{-1} \sum_m j_t(\theta; m) \right) \Big|_{\theta=\theta^\star} \right] = \mathcal{C}_t / M ,$$

and

$$\mathrm{Cov}\left[\nabla_\theta \mathcal{J}^M_{t_{1:N}}(\theta) \Big|_{\theta=\theta^\star} \right] = \frac{1}{MN}\sum_{i=1}^N \mathcal{C}_{t_i} .$$

Lemma 11. *$\nabla^2_\theta \mathcal{L}^M_{t_{1:N}}(\theta)\Big|_{\theta=\theta^\star}$ converges in probability to $\mathcal{I}_{t_{1:N}}$ as the sample size M tends to infinity, where*

$$\mathcal{I}_{t_{1:N}} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{p_i}\left[D_i(Y_i; t_{1:N}) D_i(Y_i; t_{1:N})^\top\right] ,$$

where

$$D_i(y_i; t_{1:N}) = \nabla_\theta \log p_i^\theta(y_i)\Big|_{\theta=\theta^\star} - \bar{g}^{\theta^\star}_{t_{1:N}}(y_i) \quad \textit{and} \quad \bar{g}^\theta_{t_{1:N}}(y) = \frac{\sum_{j=1}^N p_j^\theta(y) \nabla_\theta \log p_j^\theta(y)}{\sum_{j=1}^N p_j^\theta(y)} .$$

Proof.

$$\begin{aligned}
\nabla^2_\theta \mathcal{L}^M_{t_{1:N}}(\theta)\Big|_{\theta=\theta^\star} &= -\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{i=1}^{N}\left[\nabla^2_\theta \log \frac{p_i^\theta(y_i^{(m)})}{\sum_{j=1}^N p_j^\theta(y_i^{(m)})}\Big|_{\theta=\theta^\star}\right] \\
&= -\frac{1}{N}\sum_{i=1}^{N}\frac{1}{M}\sum_{m=1}^{M}\left[\nabla_\theta\left(\frac{\nabla_\theta p_i^\theta(y_i^{(m)})}{p_i^\theta(y_i^{(m)})} - \frac{\sum_j \nabla_\theta p_j^\theta(y_i^{(m)})}{\sum_j p_j^\theta(y_i^{(m)})}\right)\Big|_{\theta=\theta^\star}\right] \\
&= -\frac{1}{N}\sum_{i=1}^{N}\frac{1}{M}\sum_{m=1}^{M}\left[R(y_i^{(m)}) - K(y_i^{(m)})\right] , \qquad (56)
\end{aligned}$$

where

$$\begin{aligned}
R(y) &= \frac{\cancel{p_i^\theta(y)\textstyle\sum_j p_j^\theta(y)}\nabla_\theta(\sum_j p_j^\theta(y)\nabla_\theta p_i^\theta(y) - p_i^\theta(y)\sum_j \nabla_\theta p_j^\theta(y))}{(p_i^\theta(y)\sum_j p_j^\theta(y))\cancel{^2}}\Big|_{\theta=\theta^\star} \\
K(y) &= \frac{(\nabla_\theta(p_i^\theta(y)\sum_j p_j^\theta(y)))(\sum_j p_j^\theta(y)\nabla_\theta p_i^\theta(y) - p_i^\theta(y)\sum_j \nabla_\theta p_j^\theta(y))^\top}{(p_i^\theta(y)\sum_j p_j^\theta(y))^2}\Big|_{\theta=\theta^\star}
\end{aligned}$$

Firstly, as $M \to \infty$, the summation converges to the expectation in probability, i.e.

$$\lim_{M\to\infty}\frac{1}{M}\sum_{m=1}^{M} R(y_i^{(m)}) \xrightarrow{P} \mathbb{E}_{p_i}\left[R(Y_i)\right] \quad \text{and} \quad \lim_{M\to\infty}\frac{1}{M}\sum_{m=1}^{M} K(y_i^{(m)}) \xrightarrow{P} \mathbb{E}_{p_i}\left[K(Y_i)\right] ,$$

and we have

$$\begin{aligned}
\sum_{i=1}^{N}\mathbb{E}_{p_i}\left[R(Y_i)\right] &= \sum_{i=1}^{N}\int_{\Omega_i} p_i(y_i)\frac{\nabla_\theta(\sum_j p_j^\theta(y_i)\nabla_\theta p_i^\theta(y_i) - p_i^\theta(y_i)\sum_j \nabla_\theta p_j^\theta(y_i))\Big|_{\theta=\theta^\star}}{p_i^{\theta^\star}(y_i)\sum_j p_j^{\theta^\star}(y_i)}\mathrm{d}y_i \\
&\overset{(i)}{=} \sum_{i=1}^{N}\int_{\Omega_i}\cancel{p_i(y_i)}\frac{\nabla_\theta(\sum_j p_j^\theta(y_i)\nabla_\theta p_i^\theta(y_i) - p_i^\theta(y_i)\sum_j \nabla_\theta p_j^\theta(y_i))\Big|_{\theta=\theta^\star}}{\cancel{p_i(y_i)}\sum_j p_j(y_i)}\mathrm{d}y_i \\
&= \int_\Omega \sum_{i=1}^{N}\frac{\nabla_\theta(\sum_j p_j^\theta(y_i)\nabla_\theta p_i^\theta(y_i) - p_i^\theta(y_i)\sum_j \nabla_\theta p_j^\theta(y_i))\Big|_{\theta=\theta^\star}}{\sum_j p_j(y_i)}\mathrm{d}y_i \\
&\overset{(ii)}{=} \nabla_\theta\int_\Omega \sum_{i=1}^{N}\frac{\sum_j p_j^\theta(y_i)\nabla_\theta p_i^\theta(y_i) - p_i^\theta(y_i)\sum_j \nabla_\theta p_j^\theta(y_i)}{\sum_j p_j(y_i)}\mathrm{d}y_i\Big|_{\theta=\theta^\star} \\
&= \nabla_\theta\int_\Omega \frac{\cancel{\sum_i\sum_j p_j^\theta(y_i)\nabla_\theta p_i^\theta(y_i)} - \cancel{\sum_i p_i^\theta(y_i)\sum_j \nabla_\theta p_j^\theta(y_i)}}{\sum_j p_j(y_i)}\mathrm{d}y_i\Big|_{\theta=\theta^\star} \\
&= 0 ,
\end{aligned}$$

where (i) bases on $p_i^{\theta^\star} = p_i$ and (ii) swaps the order of differentiation and integration with mild conditions. Therefore, as $M \to \infty$, Equation (56) converges as follows in probability:

$$\lim_{M\to\infty}\nabla^2_\theta \mathcal{L}^M_{t_{1:N}}(\theta)\Big|_{\theta=\theta^\star} \xrightarrow{P} -\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{p_i}\left[-K(Y_i)\right] = \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{p_i}\left[K(Y_i)\right] .$$

Notice that

$$\nabla_\theta(p_i^\theta(y)\sum_j p_j^\theta(y)) = \underbrace{\sum_j p_j^\theta(y)\nabla_\theta p_i^\theta(y)}_{:=a_i^\theta(y)} + \underbrace{p_i^\theta(y)\sum_j \nabla_\theta p_j^\theta(y)}_{b_i^\theta(y)},$$

we could simplify

$$\begin{aligned}\frac{1}{N}\sum_{i=1}^N \mathbb{E}_{p_i}\left[K(Y_i)\right] &= \frac{1}{N}\sum_{i=1}^N \int_{\Omega_i} p_i(y_i)\frac{(a_i^{\theta^\star}(y_i)+b_i^{\theta^\star}(y_i))(a_i^{\theta^\star}(y_i)-b_i^{\theta^\star}(y_i))^\top}{(p_i^\theta(y_i)\sum_j p_j^{\theta^\star}(y_i))^2}\mathrm{d}y_i \\ &\overset{(i)}{=} \frac{1}{N}\int_\Omega \sum_{i=1}^N p_i(y_i)\frac{a_i^{\theta^\star}(y_i)a_i^{\theta^\star}(y_i)^\top - b_i^{\theta^\star}(y_i)b_i^{\theta^\star}(y_i)^\top}{p_i(y_i)^2(\sum_j p_j(y_i))^2}\mathrm{d}y_i\ , \end{aligned} \tag{57}$$

where (i) is again by $p_i^{\theta^\star} = p_i$ and

$$a_i^{\theta^\star}(y)a_i^{\theta^\star}(y)^\top = \left(\sum_j p_j(y)\nabla_\theta p_i^\theta(y)\Big|_{\theta=\theta^\star}\right)\left(\sum_j p_j(y)\nabla_\theta p_i^\theta(y)\Big|_{\theta=\theta^\star}\right)^\top \tag{58}$$

$$= p_i(y)^2\sum_k\sum_j p_k(y)p_j(y)\nabla_\theta\log p_i^\theta(y)\Big|_{\theta=\theta^\star}\left[\nabla_\theta\log p_i^\theta(y)\Big|_{\theta=\theta^\star}\right]^\top, \tag{59}$$

$$b_i^{\theta^\star}(y)b_i^{\theta^\star}(y)^\top = \left(p_i(y)\sum_j \nabla_\theta p_j^\theta(y)\Big|_{\theta=\theta^\star}\right)\left(p_i(y)\sum_j \nabla_\theta p_j^\theta(y)\Big|_{\theta=\theta^\star}\right)^\top$$

$$= p_i(y)^2\sum_k\sum_j p_k(y)p_j(y)\nabla_\theta\log p_k^\theta(y)\Big|_{\theta=\theta^\star}\left[\nabla_\theta\log p_j^\theta(y)\Big|_{\theta=\theta^\star}\right]^\top. \tag{60}$$

By plugging Equations (59) and (60) into Equation (57), we have

$$\begin{aligned}\frac{1}{N}\sum_{i=1}^N \mathbb{E}_{p_i}\left[K(Y_i)\right] &= \frac{1}{N}\int_\Omega \sum_{i=1}^N p_i(y_i)\frac{\cancel{\sum_k\sum_j p_k(y_i)p_j(y_i)}\nabla_\theta\log p_i^\theta(y_i)\Big|_{\theta=\theta^\star}\left[\nabla_\theta\log p_i^\theta(y_i)\Big|_{\theta=\theta^\star}\right]^\top}{\cancel{(\sum_j p_j(y_i))^2}} \\ &\quad - \sum_{i=1}^N \cancel{p_i(y_i)}\frac{\sum_k\sum_j p_k(y_i)p_j(y_i)\nabla_\theta\log p_k^\theta(y_i)\Big|_{\theta=\theta^\star}\left[\nabla_\theta\log p_j^{\theta^\star}(y_i)\Big|_{\theta=\theta^\star}\right]^\top}{(\sum_j p_j(y_i))^{\cancel{2}}}\mathrm{d}y_i \\ &= \frac{1}{N}\int_\Omega \sum_{i=1}^N p_i(y_i)\underbrace{\left(\nabla_\theta\log p_i^\theta(y_i)\Big|_{\theta=\theta^\star} - \bar{g}_{t_{1:N}}^{\theta^\star}(y_i)\right)}_{:=D_i(y;t_{1:N})}\left(\nabla_\theta\log p_i^\theta(y_i)\Big|_{\theta=\theta^\star} - \bar{g}_{t_{1:N}}^{\theta^\star}(y_i)\right)^\top\mathrm{d}y_i \\ &= \frac{1}{N}\sum_{i=1}^N \mathbb{E}_{p_i}\left[D_i(Y_i;t_{1:N})D_i(Y_i;t_{1:N})^\top\right] := \mathcal{I}_{t_{1:N}}\ , \end{aligned} \tag{61}$$

where

$$\bar{g}_{t_{1:N}}^{\theta^\star}(y) = \frac{\sum_{j=1}^N p_j^\theta(y)\nabla_\theta\log p_j^\theta(y)}{\sum_{j=1}^N p_j^\theta(y)} \quad\text{and}\quad D_i(y;t_{1:N}) = \nabla_\theta\log p_i^\theta(y)\Big|_{\theta=\theta^\star} - \bar{g}_{t_{1:N}}^{\theta^\star}(y)\ .$$

Hence,

$$\lim_{M\to\infty}\nabla_\theta^2\mathcal{L}_{t_{1:N}}^M(\theta)\Big|_{\theta=\theta^\star} \xrightarrow{P} \mathcal{I}_{t_{1:N}}\ .$$

□

Lemma 12. *We have* $\mathbb{E}\left[\nabla_\theta\mathcal{L}_{t_{1:N}}^M(\theta)\Big|_{\theta=\theta^\star}\right] = 0.$

Proof. First notice that

$$\mathbb{E}\left[\nabla_\theta \mathcal{L}^M_{t_{1:N}}(\theta)\Big|_{\theta=\theta^\star}\right] = \frac{1}{M}\sum_{m=1}^{M}\mathbb{E}\left[\nabla_\theta l_{t_{1:N}}(\theta;m)\Big|_{\theta=\theta^\star}\right] \overset{i.i.d.}{=} \mathbb{E}\left[\nabla_\theta l_{t_{1:N}}(\theta;m)\Big|_{\theta=\theta^\star}\right] ,$$

where the expectation is over $\otimes_{i=1}^N p_i$. We then have

$$\begin{aligned}
\mathbb{E}\left[\nabla_\theta l_{t_{1:N}}(\theta;m)\Big|_{\theta=\theta^\star}\right] &= -\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{p_i}\left[\nabla_\theta \log \frac{p_i^\theta(Y_i)}{\sum_{j=1}^N p_j^\theta(Y_i)}\Big|_{\theta=\theta^\star}\right] \\
&= -\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{p_i}\left[\frac{\nabla_\theta p_i^\theta(Y_i)\Big|_{\theta=\theta^\star}}{p_i^{\theta^\star}(Y_i)} - \frac{\sum_j \nabla_\theta p_j^\theta(Y_i)\Big|_{\theta=\theta^\star}}{\sum_j p_j^{\theta^\star}(Y_i)}\right] \\
&= -\frac{1}{N}\sum_{i=1}^{N}\left[\int_{\Omega_i} p_i(y_i)\frac{\sum_j p_j^{\theta^\star}(y_i)\nabla_\theta p_i^\theta(y_i)\Big|_{\theta=\theta^\star} - p_i^{\theta^\star}(y_i)\sum_j \nabla_\theta p_j^\theta(y_i)\Big|_{\theta=\theta^\star}}{p_i^{\theta^\star}(y_i)\sum_j p_j^{\theta^\star}(y_i)}\mathrm{d}y_i\right] .
\end{aligned}$$

By using the extended support Ω and the optimality $p_i^{\theta^\star} = p_i$, we have

$$\begin{aligned}
\mathbb{E}\left[\nabla_\theta l_{t_{1:N}}(\theta;m)\Big|_{\theta=\theta^\star}\right] &= -\frac{1}{N}\int_\Omega \sum_{i=1}^{N} \cancel{p_i(y_i)} \frac{\sum_j p_j(y_i)\nabla_\theta p_i^\theta(y_i)\Big|_{\theta=\theta^\star} - p_i(y_i)\sum_j \nabla_\theta p_j^\theta(y_i)\Big|_{\theta=\theta^\star}}{\cancel{p_i(y_i)}\sum_j p_j(y_i)}\mathrm{d}y_i \\
&= -\frac{1}{N}\int_\Omega \frac{\cancel{\sum_i \sum_j p_j(y_i)\nabla_\theta p_i^\theta(y_i)\Big|_{\theta=\theta^\star}} - \cancel{\sum_i p_i(y_i)\sum_j \nabla_\theta p_j^\theta(y_i)\Big|_{\theta=\theta^\star}}}{\sum_j p_j(y_i)}\mathrm{d}y_i \\
&= 0 .
\end{aligned}$$

□

Lemma 13. *The covariance* $\mathrm{Cov}\left[\nabla_\theta \mathcal{L}^M_{t_{1:N}}(\theta^\star)\Big|_{\theta=\theta^\star}\right]$ *is*

$$\frac{1}{M}\left(\frac{1}{N}\mathcal{I}_{t_{1:N}} - \frac{1}{N^2}\sum_i \mathbb{E}_{p_i}\left[D_i(Y_i)\right]\mathbb{E}_{p_i}\left[D_i(Y_i)\right]^\top\right) ,$$

where $D_i(y_i) = \nabla_\theta \log p_i^\theta(y_i)\Big|_{\theta=\theta^\star} - \bar{g}_{\theta^\star}(y_i)$, $\bar{g}_{\theta^\star}(y_i)$ *is defined in Lemma 11, and the expectations are taken over* p_i.

Proof. As $\mathbb{E}\left[\nabla_\theta \mathcal{L}^M_{t_{1:N}}(\theta)\Big|_{\theta=\theta^\star}\right] = 0$, we have

$$\begin{aligned}
\mathrm{Cov}\left[\nabla_\theta \mathcal{L}^M_{t_{1:N}}(\theta)\Big|_{\theta=\theta^\star}\right] &= \mathbb{E}\left[\nabla_\theta \mathcal{L}^M_{t_{1:N}}(\theta)\Big|_{\theta=\theta^\star}\left(\nabla_\theta \mathcal{L}^M_{t_{1:N}}(\theta^\star)\Big|_{\theta=\theta}\right)^\top\right] \\
&\overset{i.i.d.}{=} \frac{1}{M}\mathbb{E}\left[\nabla_\theta l_{t_{1:N}}(\theta;m)\Big|_{\theta=\theta^\star}\left(\nabla_\theta l_{t_{1:N}}(\theta;m)\Big|_{\theta=\theta^\star}\right)^\top\right] ,
\end{aligned}$$

where

$$\nabla_\theta l_{t_{1:N}}(\theta^\star;m) = -\frac{1}{N}\sum_{i=1}^{N}\left(\underbrace{\nabla_\theta \log p_i^\theta(Y_i)\Big|_{\theta=\theta^\star} - \bar{g}_{\theta^\star}(Y_i)}_{D_i(y_i)}\right) .$$

Hence,

$$\begin{aligned}
&\mathbb{E}\left[\nabla_\theta l_{t_{1:N}}(\theta;m)\Big|_{\theta=\theta^\star}\left(\nabla_\theta l_{t_{1:N}}(\theta;m)\Big|_{\theta=\theta^\star}\right)^\top\right] \\
&= \frac{1}{N^2}\mathbb{E}\left[\left(\sum_i D_i(Y_i)\right)\left(\sum_i D_i(Y_i)\right)^\top\right] \\
&\overset{(i)}{=} \frac{1}{N^2}\left(\sum_i \mathbb{E}_{p_i}\left[D_i(Y_i)D_i(Y_i)^\top\right] + \sum_{i\neq j}\mathbb{E}_{p_i}\left[D_i(Y_i)\right]\mathbb{E}_{p_j}\left[D_j(Y_j)\right]^T\right) ,
\end{aligned}$$

where (i) bases on the i.i.d. assumption and

$$\sum_i \mathbb{E}_{p_i}\left[D_i(Y_i)D_i(Y_i)^\top\right] = N\mathcal{I}_{t_{1:N}} .$$

To explore the second term, first notice that

$$\sum_i \mathbb{E}_{p_i}\left[D_i(Y_i)\right] = \int_\Omega \sum_i \left(\nabla_\theta p_i^\theta(y_i)\Big|_{\theta=\theta^\star} - p_i(y_i)\frac{\sum_j \nabla_\theta p_j^\theta(y_i)\Big|_{\theta=\theta^\star}}{\sum_j p_j(y_i)}\right)\mathrm{d}y_i = 0 .$$

Hence,

$$\begin{aligned}
0 &= \left(\sum_i \mathbb{E}_{p_i}\left[D_i(Y_i)\right]\right)\left(\sum_i \mathbb{E}_{p_i}\left[D_i(Y_i)\right]\right)^\top \\
&= \sum_i \mathbb{E}_{p_i}\left[D_i(Y_i)\right]\mathbb{E}_{p_i}\left[D_i(Y_i)\right]^\top + \sum_{i\neq j}\mathbb{E}_{p_i}\left[D_i(Y_i)\right]\mathbb{E}_{p_j}\left[D_j(X_j)\right]^\top ,
\end{aligned}$$

that means

$$\sum_{i\neq j}\mathbb{E}_{p_i}\left[D_i(Y_i)\right]\mathbb{E}_{p_j}\left[D_j(X_j)\right]^\top = -\sum_i \mathbb{E}_{p_i}\left[D_i(Y_i)\right]\mathbb{E}_{p_i}\left[D_i(Y_i)\right]^\top ,$$

and therefore

$$\begin{aligned}
\mathrm{Cov}\left[\nabla_\theta \mathcal{L}^M_{t_{1:N}}(\theta)\Big|_{\theta=\theta^\star}\right] &= \frac{1}{M}\mathbb{E}\left[\nabla_\theta l_{t_{1:N}}(\theta;m)\Big|_{\theta=\theta^\star}\left(\nabla_\theta l_{t_{1:N}}(\theta;m)\Big|_{\theta=\theta^\star}\right)^\top\right] \\
&= \frac{1}{M}\frac{1}{N^2}\left(N\mathcal{I}_{t_{1:N}} - \sum_i \mathbb{E}_{p_i}\left[D_i(Y_i)\right]\mathbb{E}_{p_i}\left[D_i(Y_i)\right]^\top\right) .
\end{aligned}$$

□

Lemma 14 (Asymptotic normality). *$\sqrt{M}(\hat{\theta}^M_{t_{1:N}} - \theta^\star_{t_{1:N}})$ is asymptotically normal with zero mean and covariance matrix $\Sigma_{t_{1:N}}$*

$$\Sigma_{t_{1:N}} = \frac{1}{N}\left(\mathcal{I}_{t_{1:N}} + \mathcal{P}_{t_{1:N}}\right)^{-1}\left(\mathcal{I}_{t_{1:N}} - \frac{1}{N}\sum_i \mathbb{E}_{p_i}\left[D_i(Y_i)\right]\mathbb{E}_{p_i}\left[D_i(Y_i)\right]^\top + \mathcal{C}_{t_{1:N}}\right)\left(\mathcal{I}_{t_{1:N}} + \mathcal{P}_{t_{1:N}}\right)^{-1} ,$$

if $\mathcal{I}_{t_{1:N}}$ is full rank and $\mathcal{P}_{t_{1:N}}$ is positive-definite.

Proof. We first Taylor expand $\nabla_\theta\left(\mathcal{L}^M_{t_{1:N}}(\theta) + \mathcal{J}^M_{t_{1:N}}(\theta)\right)$ at $\hat{\theta}^M_{t_{1:N}}$ around $\theta^\star$:

$$\begin{aligned}
0 = \nabla_\theta\left(\mathcal{L}^M_{t_{1:N}}(\theta) + \mathcal{J}^M_{t_{1:N}}(\theta)\right)\Big|_{\theta=\hat{\theta}^M_{t_{1:N}}} &\approx \left(\nabla_\theta \mathcal{L}^M_{t_{1:N}}(\theta) + \nabla_\theta \mathcal{J}^M_{t_{1:N}}(\theta)\right)\Big|_{\theta=\theta^\star} \\
&\quad + \left(\nabla^2_\theta \mathcal{L}^M_{t_{1:N}}(\theta) + \nabla^2_\theta \mathcal{J}^M_{t_{1:N}}(\theta)\right)\Big|_{\theta=\theta^\star}(\hat{\theta}^M_{1:N} - \theta^\star) . \quad (62)
\end{aligned}$$

Therefore, we have

$$\sqrt{M}(\hat{\theta}^M_{t_{1:N}} - \theta^\star) \approx -\left[\left(\nabla^2_\theta \mathcal{L}^M_{t_{1:N}}(\theta) + \nabla^2_\theta \mathcal{J}^M_{t_{1:N}}(\theta)\right)\Big|_{\theta=\theta^\star}\right]^{-1}\sqrt{M}\left(\nabla_\theta \mathcal{L}^M_{t_{1:N}}(\theta) + \nabla_\theta \mathcal{J}^M_{t_{1:N}}(\theta)\right)\Big|_{\theta=\theta^\star} .$$

By assumption 8 and lemma 11, $\nabla_\theta^2 \mathcal{L}^M_{t_{1:N}}(\theta)\Big|_{\theta=\theta^\star} \xrightarrow{P} \mathcal{I}_{t_{1:N}}$ and $\nabla_\theta^2 \mathcal{J}^M_{t_{1:N}}(\theta)\Big|_{\theta=\theta^\star} \xrightarrow{P} \mathcal{P}_{t_{1:N}}$ for large sample size M. Using assumptions 9 and 10 and lemmas 12 and 13, we see that

$$\sqrt{M}\left(\nabla_\theta \mathcal{L}^M_{t_{1:N}}(\theta) + \nabla_\theta \mathcal{J}^M_{t_{1:N}}(\theta)\right)\Big|_{\theta=\theta^\star},$$

converges in distribution to a Gaussian distribution with mean zero and covariance

$$\begin{aligned} &M\left(\mathrm{Cov}\left[\nabla_\theta \mathcal{L}_{t_{1:N}}(\theta)\Big|_{\theta=\theta^\star}\right] + \mathrm{Cov}\left[\nabla_\theta \mathcal{J}_{t_{1:N}}(\theta)\Big|_{\theta=\theta^\star}\right]\right) \\ &= \frac{1}{N}\mathcal{I}_{t_{1:N}} - \frac{1}{N^2}\sum_i \mathbb{E}_{p_i}\left[D_i(Y_i)\right]\mathbb{E}_{p_i}\left[D_i(Y_i)\right]^\top + \frac{1}{N}\sum_i \mathcal{C}_{t_i}. \end{aligned}$$

In meanwhile, since $\mathcal{I}_{t_{1:N}}$ is positive-semidefinite, by using the assumption that it has full rank, we have $\mathcal{I}_{t_{1:N}}$ is a positive-definite matrix. By using the assumption that $\mathcal{P}_{t_{1:N}}$ is positive-definite, we have $\mathcal{I}_{t_{1:N}} + \mathcal{P}_{t_{1:N}}$ is positive-definite and therefore invertible.

Combing the above results, we could show that $\sqrt{M}(\hat{\theta}^M_{t_{1:N}} - \theta^\star)$ converges in distribution to a Gaussian distribution of zero mean and covariance $\Sigma_{t_{1:N}}$,

$$\Sigma_{t_{1:N}} = \frac{1}{N}\left(\mathcal{I}_{t_{1:N}} + \mathcal{P}_{t_{1:N}}\right)^{-1}\left(\mathcal{I}_{t_{1:N}} - \frac{1}{N}\sum_i \mathbb{E}_{p_i}\left[D_i(Y_i)\right]\mathbb{E}_{p_i}\left[D_i(Y_i)\right]^\top + \sum_i \mathcal{C}_{t_i}\right)\left(\mathcal{I}_{t_{1:N}} + \mathcal{P}_{t_{1:N}}\right)^{-1}.$$

□

E.3.2 Proof of Proposition 3

For clarity, we present the general `DiffCLF` loss defined in Equation (10) and the DSM loss defined in Equation (7) as references,

$$\begin{aligned} \mathcal{L}_{\mathrm{clf}}(\theta; N) &= -\mathbb{E}_{t_{1:N}}\frac{1}{N}\sum_{i=1}^N \mathbb{E}_{p_i}\left[\log \frac{p_i^\theta(Y_i)}{\sum_j p_j^\theta(Y_i)}\right], \\ \mathcal{L}_{\mathrm{DSM}}(\theta) &= \mathbb{E}_t \mathbb{E}_{q_t}\mathbb{E}_{p_t(\cdot|\tilde{X}_t)}\left[\left\|\nabla \log p_t^\theta(\tilde{Y}_t) - \nabla \log p_t(\tilde{Y}_t|\tilde{X}_t)\right\|^2\right], \end{aligned}$$

and the empirical losses are

$$\mathcal{L}^M_{\mathrm{clf}}(\theta; N) = \frac{1}{M}\sum_{m=1}^M l(\theta; m), \quad l(\theta; m) = \frac{1}{N}\sum_{i=1}^N \log \frac{p^\theta_{t_i^{(m)}}(Y_i^{(m)})}{\sum_j p^\theta_{t_j^{(m)}}(Y_i^{(m)})}, \tag{63}$$

$$\mathcal{L}^M_{\mathrm{DSM}}(\theta) = \frac{1}{M}\sum_{m=1}^M j(\theta; m), \quad j(\theta; m) = \left\|\nabla \log p^\theta_{\tilde{t}^{(m)}}(\tilde{Y}_{\tilde{t}^{(m)}}) - \nabla \log p_t(\tilde{Y}_{\tilde{t}^{(m)}}|\tilde{X}_{\tilde{t}^{(m)}})\right\|^2, \tag{64}$$

where $t^{(m)}_{1:N} \sim U([0,T])^N$, $Y_i^{(m)} \sim p_{t_i^{(m)}}$, $\tilde{t}^{(m)} \sim U([0,T])$, $\tilde{X}_{\tilde{t}^{(m)}} \sim q_{\tilde{t}^{(m)}}$, $\tilde{Y}_{\tilde{t}^{(m)}} \sim p_{\tilde{t}^{(m)}}(\cdot|\tilde{X}_{\tilde{t}^{(m)}})$. All samples are independent to each other, except for $\tilde{X}_{\tilde{t}^{(m)}}$ and $\tilde{Y}_{\tilde{t}^{(m)}}$.

Define the population joint loss and empirical joint loss as

$$\mathcal{L}_{\mathrm{joint}}(\theta; N) = \mathcal{L}_{\mathrm{DSM}}(\theta) + \mathcal{L}_{\mathrm{clf}}(\theta; N), \tag{65}$$

$$\mathcal{L}^M_{\mathrm{joint}}(\theta; N) = \mathcal{L}^M_{\mathrm{DSM}}(\theta) + \mathcal{L}^M_{\mathrm{clf}}(\theta; N). \tag{66}$$

Proposition 3 (Formal). *Let $\hat{\theta}^M$ be the optimal parameter learned with the empirical joint loss (Equation (66)). If conditions (a) to (c) are fulfilled, i.e.*

(a) $\sup_\theta |\mathcal{L}^M_{\mathrm{joint}}(\theta) - \mathcal{L}_{\mathrm{joint}}(\theta)| \xrightarrow{P} 0$ *as* $M \to \infty$,

(b) The matrix $\mathcal{I} = \mathbb{E}_{t_{1:N}}\left[\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{p_i}\left[D_i(Y_i;t_{1:N})D_i(Y_i;t_{1:N})^\top\right]\right]$ *has full rank, where*

$$D_i(y_i;t_{1:N}) = \nabla_\theta \log p_i^\theta(y_i)\Big|_{\theta=\theta^\star} - \bar{g}^{\theta^\star}_{t_{1:N}}(y_i) \quad \text{and} \quad \bar{g}^{\theta^\star}_{t_{1:N}}(y) = \frac{\sum_{j=1}^{N} p_j(y)\nabla_\theta \log p_j^\theta(y)\Big|_{\theta=\theta^\star}}{\sum_{j=1}^{N} p_j(y)} .$$

(c) The matrices $(\mathcal{P}_t)_t$ *are positive-definite, where*

$$\lim_{M\to\infty}\nabla_\theta^2\left(\frac{1}{M}\sum_m \|\nabla \log p_t^\theta(\tilde{Y}_t^{(m)}) - \nabla \log p_{t^{(m)}}(\tilde{Y}_t^{(m)}|\tilde{X}_t^{(m)})\|^2\right) \xrightarrow{P} \mathcal{P}_t ,$$

with $\tilde{X}_t^{(m)} \overset{i.i.d.}{\sim} q_t$ *and* $\tilde{Y}_t^{(m)} \overset{i.i.d.}{\sim} p_t(\cdot|\tilde{X}_t^{(m)})$.

Then $\hat{\theta}^M$ *has the following properties*

(1) ***(Consistency)*** $\hat{\theta}^M$ *is a consistent estimator of* $\theta^\star$

(2) ***(Asymptotically Normality)*** $\sqrt{M}(\hat{\theta}^M - \theta^\star)$ *is asymptotically normal with zero mean and covariance matrix* Σ*,*

$$\Sigma = (\mathcal{I}+\mathcal{P})^{-1}\left(\underbrace{\frac{1}{N}\mathcal{I} - \frac{1}{N^2}\mathbb{E}_{t_{1:N}}\left[\sum_i \mathbb{E}_{p_{t_i}}\left[D_i(Y_i;t_{1:N})\right]\mathbb{E}_{p_{t_i}}\left[D_i(Y_i;t_{1:N})\right]^\top\right]}_{=\frac{1}{N^2}\mathbb{E}_{t_{1:N}}\sum_i \mathrm{Cov}_{p_{t_i}}(D_i(Y_i;t_{1:N}))} + \mathcal{C}\right)(\mathcal{I}+\mathcal{P})^{-1} , \tag{67}$$

where $D_i(y_i;t_{1:N})$ *is defined in condition (b),* $\mathcal{P} = \mathbb{E}_t[\mathcal{P}_t]$*, and* $\mathcal{C} = \mathbb{E}_t[\mathcal{C}_t]$ *with*

$$\mathcal{C}_t = \mathrm{Cov}\left[\nabla_\theta\left(\left\|\nabla \log p_t^\theta(\tilde{Y}_t) - \nabla \log p_t(\tilde{Y}_t|\tilde{X}_t)\right\|^2\right)\right],$$

with $\tilde{X}_t \sim q_t$ *and* $\tilde{Y}_t \sim p_t(\cdot|\tilde{X}_t)$.

(3) ***(Error Bound)*** *For large sample sizes* M*,* $\mathbb{E}[\|\hat{\theta}^M - \theta^\star\|^2] \approx \mathrm{tr}(\Sigma)/M$.

Proof. Firstly, recall that Proposition 1 shows that $\theta^\star$ is a unique minimizer. Following standard procedure in M-estimators, and with condition (a), the consistency is trivial to prove.

To show the asymptotical normality and the error bound, we require the following quantities evaluated at the optimality $\theta^\star$:

(i) the asymptotic Hessian of loss,

$$\lim_{M\to\infty}\nabla_\theta^2\mathcal{L}^M_{\text{joint}}(\theta)\Big|_{\theta=\theta^\star} = \lim_{M\to\infty}\nabla_\theta^2\mathcal{L}^M_{\text{DSM}}(\theta)\Big|_{\theta=\theta^\star} + \lim_{M\to\infty}\nabla_\theta^2\mathcal{L}^M_{\text{clf}}(\theta)\Big|_{\theta=\theta^\star} ,$$

(ii) the mean of gradient of loss,

$$\mathbb{E}\left[\nabla_\theta\mathcal{L}^M_{\text{joint}}(\theta)\Big|_{\theta=\theta^\star}\right] = \mathbb{E}\left[\nabla_\theta\mathcal{L}^M_{\text{DSM}}(\theta)\Big|_{\theta=\theta^\star}\right] + \mathbb{E}\left[\nabla_\theta\mathcal{L}^M_{\text{clf}}(\theta)\Big|_{\theta=\theta^\star}\right] ,$$

(iii) the covariance of gradient of loss,

$$\mathrm{Cov}\left[\nabla_\theta\mathcal{L}^M_{\text{joint}}(\theta)\Big|_{\theta=\theta^\star}\right] = \mathrm{Cov}\left[\nabla_\theta\mathcal{L}^M_{\text{DSM}}(\theta)\Big|_{\theta=\theta^\star}\right] + \mathrm{Cov}\left[\nabla_\theta\mathcal{L}^M_{\text{clf}}(\theta)\Big|_{\theta=\theta^\star}\right] ,$$

(i) Following the proof of Lemma 14, we have

$$\lim_{M\to\infty}\nabla_\theta^2\mathcal{L}^M_{\text{clf}}(\theta)\Big|_{\theta=\theta^\star} \xrightarrow{P} \mathbb{E}_{t_{1:N}}[\mathcal{I}_{t_{1:N}}] := \mathcal{I} \quad \text{and} \quad \lim_{M\to\infty}\nabla_\theta^2\mathcal{L}^M_{\text{DSM}}(\theta)\Big|_{\theta=\theta^\star} \xrightarrow{P} \mathbb{E}_t[\mathcal{P}_t] := \mathcal{P} .$$

By the condition (b), the semi-positive definite matrix $\mathcal{I}$ has full rank, meaning that it is positive-definite. By the condition (c), $\mathcal{P}$ is a weighted sum of positive-definite matrices and therefore it is positive-definite as well. Combing these, $\mathcal{I}+\mathcal{P}$ is positive-definite, and therefore is invertible.

(ii) By leveraging the i.i.d. condition for Monte Carlo samples, we have

$$\mathbb{E}\left[\nabla_\theta \mathcal{L}^M_{\text{clf}}(\theta)\Big|_{\theta=\theta^\star}\right] = \mathbb{E}\left[\nabla_\theta l(\theta;m)\Big|_{\theta=\theta^\star}\right] = \mathbb{E}_{t_{1:N}}\mathbb{E}_{Y^{(m)}_{1:N}|t_{1:N}}\left[\nabla_\theta l(\theta;m)\Big|_{\theta=\theta^\star}\right] ,$$
$$\mathbb{E}\left[\nabla_\theta \mathcal{L}^M_{\text{DSM}}(\theta)\Big|_{\theta=\theta^\star}\right] = \mathbb{E}\left[\nabla_\theta j(\theta;m)\Big|_{\theta=\theta^\star}\right] = \mathbb{E}_t\mathbb{E}_{\tilde{X}_t}\mathbb{E}_{\tilde{Y}|\tilde{X}}\left[\nabla_\theta j(\theta;m)\Big|_{\theta=\theta^\star}\right] ,$$

where $\mathbb{E}_{Y^{(m)}_{1:N}|t_{1:N}}\left[\nabla_\theta l(\theta;m)\Big|_{\theta=\theta^\star}\right] = \mathbb{E}_{\tilde{X}_t|t}\mathbb{E}_{\tilde{Y}_t|\tilde{X}_t}\left[\nabla_\theta j(\theta;m)\Big|_{\theta=\theta^\star}\right] = 0$ is given by Lemma 12 and assumption 9. Hence, $\mathbb{E}\left[\nabla_\theta \mathcal{L}^M_{\text{joint}}(\theta)\Big|_{\theta=\theta^\star}\right] = 0$.

(iii) Again, we apply the i.i.d. condition,

$$\text{Cov}\left[\nabla_\theta \mathcal{L}^M_{\text{clf}}(\theta)\Big|_{\theta=\theta^\star}\right] = \frac{1}{M}\text{Cov}\left[\nabla_\theta l(\theta;m)\Big|_{\theta=\theta^\star}\right] = \frac{1}{M}\mathbb{E}\left[\nabla_\theta l(\theta;m)\Big|_{\theta=\theta^\star}\left(\nabla_\theta l(\theta;m)\Big|_{\theta=\theta^\star}\right)^\top\right],$$
$$\text{Cov}\left[\nabla_\theta \mathcal{L}^M_{\text{DSM}}(\theta)\Big|_{\theta=\theta^\star}\right] = \frac{1}{M}\text{Cov}\left[\nabla_\theta j(\theta;m)\Big|_{\theta=\theta^\star}\right] = \frac{1}{M}\mathbb{E}\left[\nabla_\theta j(\theta;m)\Big|_{\theta=\theta^\star}\left(\nabla_\theta j(\theta;m)\Big|_{\theta=\theta^\star}\right)^\top\right] ,$$

where we could split the expectation again [12]

$$\frac{1}{M}\mathbb{E}\left[\nabla_\theta l(\theta;m)\Big|_{\theta=\theta^\star}\left(\nabla_\theta l(\theta;m)\Big|_{\theta=\theta^\star}\right)^\top\right] = \frac{1}{M}\mathbb{E}_{t_{1:N}}\mathbb{E}_{Y^{(m)}_{1:N}|t_{1:N}}\left[\nabla_\theta l(\theta;m)\Big|_{\theta=\theta^\star}\left(\nabla_\theta l(\theta;m)\Big|_{\theta=\theta^\star}\right)^\top\right]$$
$$\overset{(a)}{=} \frac{1}{M}\mathbb{E}_{t_{1:N}}\left[\frac{1}{N^2}\left(N\mathcal{I}_{t_{1:N}} - \sum_i \mathbb{E}_{p_{t_i}}\left[D_i(Y_i)\right]\mathbb{E}_{p_{t_i}}\left[D_i(Y_i)\right]^\top\right)\right],$$
$$\frac{1}{M}\mathbb{E}\left[\nabla_\theta j(\theta;m)\Big|_{\theta=\theta^\star}\left(\nabla_\theta j(\theta;m)\Big|_{\theta=\theta^\star}\right)^\top\right] = \frac{1}{M}\mathbb{E}_t\mathbb{E}_{\tilde{X}^{(m)}_t|t}\mathbb{E}_{\tilde{Y}^{(m)}_t|\tilde{X}^{(m)}_t}\left[\nabla_\theta j(\theta;m)\Big|_{\theta=\theta^\star}\left(\nabla_\theta j(\theta;m)\Big|_{\theta=\theta^\star}\right)^\top\right]$$
$$\overset{(b)}{=} \frac{1}{M}\mathbb{E}_t\left[\mathcal{C}_t\right] := \frac{1}{M}\mathcal{C} ,$$

where (a) is given by Lemma 13 and (b) is by Assumption 10. Hence,

$$\text{Cov}\left[\sqrt{M}\nabla_\theta \mathcal{L}^M_{\text{joint}}(\theta^\star)\right] = M\,\text{Cov}\left[\nabla_\theta \mathcal{L}^M_{\text{clf}}(\theta)\Big|_{\theta=\theta^\star}\right] + M\,\text{Cov}\left[\nabla_\theta \mathcal{L}^M_{\text{DSM}}(\theta)\Big|_{\theta=\theta^\star}\right]$$
$$= \frac{1}{N}\mathcal{I} - \frac{1}{N^2}\mathbb{E}_{t_{1:N}}\left[\sum_i \mathbb{E}_{p_{t_i}}\left[D_i(Y_i)\right]\mathbb{E}_{p_{t_i}}\left[D_i(Y_i)\right]^\top\right] + \mathcal{C} .$$

Combining (i)-(iii) and according to Taylor expansion which gives

$$\sqrt{M}(\hat{\theta}^M - \theta^\star) \approx -\left[\nabla^2_\theta \mathcal{L}^M_{\text{joint}}(\theta)\Big|_{\theta=\theta^\star}\right]^{-1}\left(\sqrt{M}\nabla_\theta \mathcal{L}^M_{\text{joint}}(\theta)\Big|_{\theta=\theta^\star}\right) , \tag{68}$$

we have

$$\mathbb{E}\left[\sqrt{M}(\hat{\theta}^M - \theta^\star)\right] = 0 \quad \text{and} \quad \lim_{M\to\infty}\text{Cov}\left[\sqrt{M}(\hat{\theta}^M - \theta^\star)\right] \xrightarrow{P} \Sigma ,$$

where

$$\Sigma := (\mathcal{I}+\mathcal{P})^{-1}\left(\frac{1}{N}\mathcal{I} - \frac{1}{N^2}\mathbb{E}_{t_{1:N}}\left[\sum_i \mathbb{E}_{p_{t_i}}\left[D_i(Y_i)\right]\mathbb{E}_{p_{t_i}}\left[D_i(Y_i)\right]^\top\right] + \mathcal{C}\right)(\mathcal{I}+\mathcal{P})^{-1} ,$$

and notice that

$$\frac{1}{N}\mathcal{I} - \frac{1}{N^2}\mathbb{E}_{t_{1:N}}\left[\sum_i \mathbb{E}_{p_{t_i}}\left[D_i(Y_i)\right]\mathbb{E}_{p_{t_i}}\left[D_i(Y_i)\right]^\top\right] = \frac{1}{N^2}\mathbb{E}_{t_{1:N}}\left[\sum_i \text{Cov}_{p_{t_i}}\left[D_i(Y_i)\right]\right] .$$

That is, $\sqrt{M}(\hat{\theta}^M - \theta^\star)$ in probability converges to a Gaussian distribution with zero mean and covariance Σ. □

[12] For clarity, we write $D_i(y;t_{1:N})$ as $D_i(y)$ in the proof.

Proposition 15. *Let's treat the information matrix $\mathcal{I}$ and the asymptotic covariance Σ defined in Proposition 3 as functions depending on N, i.e.*

$$\mathcal{I}(N) = \mathbb{E}_{t_{1:N}}\left[\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{p_{t_i}}\left[D_i(Y_i;t_{1:N})D_i(Y_i;t_{1:N})^\top\right]\right],$$

$$\Sigma(N) = (\mathcal{I}(N)+\mathcal{P})^{-1}\left(\frac{1}{N}\mathcal{I}(N) - \frac{1}{N^2}\mathbb{E}_{t_{1:N}}\left[\sum_i \mathbb{E}_{p_{t_i}}\left[D_i(Y_i;t_{1:N})\right]\mathbb{E}_{p_{t_i}}\left[D_i(Y_i;t_{1:N})\right]^\top\right] + \mathcal{C}\right)(\mathcal{I}(N)+\mathcal{P})^{-1}\ .$$

By assuming that $\int_0^T \mathbb{E}_{p_t}\left[\|\nabla_\theta \log p_t^{\theta^\star}(X)\|^2\right]\,\mathrm{d}t < \infty$, then

$$\lim_{N\to\infty}\mathcal{I}(N) = \mathcal{I}_\infty := \int_0^T \mathbb{E}_{p_t}\left[D_t(y)D_t(y)^\top\right]\mathrm{d}t\ ,$$

where $D_t(y) = \nabla_\theta \log p_t^\theta(y)\Big|_{\theta=\theta^\star} - c(y)$ and $c(y) = \frac{\int_0^T p_t(y)\nabla_\theta \log p_t^\theta(y)\Big|_{\theta=\theta^\star}\mathrm{d}t}{\int_0^T p_t(y)\mathrm{d}t}$. By further assuming $\mathcal{I}_\infty$ is finite and non-singular, then

$$\lim_{N\to\infty} N\left(\Sigma(N) - (\mathcal{I}_\infty-\mathcal{P})^{-1}\mathcal{C}\,(\mathcal{I}_\infty-\mathcal{P})^{-1}\right)$$
$$= (\mathcal{I}_\infty-\mathcal{P})^{-1}\left(\int_0^T \left(\mathbb{E}_{p_t}\left[D_t(Y_t)\right]\right)\left(\mathbb{E}_{p_t}\left[D_t(Y_t)\right]\right)^\top \mathrm{d}t\right)(\mathcal{I}_\infty-\mathcal{P})^{-1}\ .$$

Proof. First note that, for each fixed x, by the strong law of large numbers applied to the i.i.d. sequence $\{t_j\}_{j\geq 1}$ with $t_j \sim \mathrm{Unif}[0,T]$,

$$\frac{1}{N}\sum_{j=1}^{N} p_j(y)\nabla_\theta \log p_j^\theta(y)\Big|_{\theta=\theta^\star} \xrightarrow[N\to\infty]{a.s.} \mathbb{E}_t\left[p_t(y)\nabla_\theta \log p_t^\theta(y)\Big|_{\theta=\theta^\star}\right] = \frac{1}{T}\int_0^T p_t(y)\nabla_\theta \log p_t^\theta(y)\Big|_{\theta=\theta^\star}\mathrm{d}t\ ,$$

and

$$\frac{1}{N}\sum_{j=1}^{N} p_j(y) \xrightarrow[N\to\infty]{a.s.} \mathbb{E}_t[p_t(y)] = \frac{1}{T}\int_0^T p_t(y)\,\mathrm{d}t\ ,$$

so that, whenever the denominator is non-zero,

$$\bar{g}^{\theta^\star}_{t_{1:N}}(y) = \frac{\sum_{j=1}^N p_j(y)\nabla_\theta \log p_j^{\theta^\star}(y)\Big|_{\theta=\theta^\star}}{\sum_{j=1}^N p_j(y)} \xrightarrow[N\to\infty]{a.s.} \frac{\int_0^T p_t(y)\nabla_\theta \log p_t^\theta(y)\Big|_{\theta=\theta^\star}\mathrm{d}t}{\int_0^T p_t(y)\,\mathrm{d}t} = c(y)\ .$$

Consequently,

$$D_i(y;t_{1:N}) = \nabla_\theta \log p_i^{\theta^\star}(y)\Big|_{\theta=\theta^\star} - \bar{g}^{\theta^\star}_{t_{1:N}}(y) \xrightarrow[N\to\infty]{a.s.} D_{t_i}(y) := \nabla_\theta \log p_{t_i}^{\theta^\star}(y)\Big|_{\theta=\theta^\star} - c(y)\ .$$

By the assumed square integrability of $\nabla_\theta \log p_t^{\theta^\star}\Big|_{\theta=\theta^\star}$ and the definition of $c(y)$, the random matrices

$$D_i(Y;t_{1:N})D_i(Y;t_{1:N})^\top$$

are dominated in L^1 by an integrable function that does not depend on N. Hence, by dominated convergence,

$$\lim_{N\to\infty}\mathbb{E}_{p_i}\left[D_i(Y_i;t_{1:N})D_i(Y_i;t_{1:N})^\top\right] = \mathbb{E}_{p_{t_i}}\left[D_{t_i}(Y_i)D_{t_i}(Y_i)^\top\right] \quad \text{a.s. in } t_{1:N}\ .$$

Taking the average over i and then expectation with respect to $t_{1:N}$, and applying again the law of large numbers in i together with dominated convergence, yields

$$\lim_{N\to\infty}\mathcal{I}(N) = \mathbb{E}_t\left[\mathbb{E}_{p_t}\left[D_t(Y_t)D_t(Y_t)^\top\right]\right] = \frac{1}{T}\int_0^T \mathbb{E}_{p_t}\left[D_t(Y_t)D_t(Y_t)^\top\right]\mathrm{d}t = \mathcal{I}_\infty\ .$$

The same argument applies to the means m_i: since

$$m_i = \mathbb{E}_{p_i}[D_i(Y_i; t_{1:N})] \xrightarrow[N\to\infty]{} \mathbb{E}_{p_{t_i}}[D_{t_i}(Y_i)] \,,$$

and the latter are square integrable in t by assumption, we obtain

$$\begin{aligned}\lim_{N\to\infty} \mathbb{E}_{t_{1:N}}\left[\frac{1}{N}\sum_{i=1}^{N} m_i m_i^\top\right] &= \mathbb{E}_t\left[\left(\mathbb{E}_{p_t}[D_t(Y_t)]\right)\left(\mathbb{E}_{p_t}[D_t(Y_t)]\right)^\top\right] \\ &= \frac{1}{T}\int_0^T \left(\mathbb{E}_{p_t}[D_t(Y_t)]\right)\left(\mathbb{E}_{p_t}[D_t(Y_t)]\right)^\top \mathrm{d}t \\ &= \mathcal{Q}_\infty \,.\end{aligned}$$

Using the definition of $\Sigma(N)$, we can write

$$\begin{aligned}&N\left(\Sigma(N) - (\mathcal{I}_\infty - \mathcal{P})^{-1}\,\mathcal{C}\,(\mathcal{I}_\infty - \mathcal{P})^{-1}\right) \\ &= (\mathcal{I}(N)+\mathcal{P})^{-1}\left(\mathcal{I}(N) - \frac{1}{N}\mathbb{E}_{t_{1:N}}\left[\sum_i m_i m_i^\top\right] + N\mathcal{C}\right)(\mathcal{I}(N)+\mathcal{P})^{-1} - N\,(\mathcal{I}_\infty - \mathcal{P})^{-1}\,\mathcal{C}\,(\mathcal{I}_\infty - \mathcal{P})^{-1} \,.\end{aligned}$$

By the first part of the proof, $\mathcal{I}(N) \to \mathcal{I}_\infty$ and $\frac{1}{N}\mathbb{E}_{t_{1:N}}\left[\sum_i m_i m_i^\top\right] \to \mathcal{Q}_\infty$. Since $\mathcal{I}_\infty$ is non-singular, the matrix inversion map is continuous at $\mathcal{I}_\infty$, and thus $\mathcal{I}(N)^{-1} \to \mathcal{I}_\infty^{-1}$. Together with the convergence of the mean outer products to $\mathcal{J}_\infty$, we obtain

$$\lim_{N\to\infty} N\left(\Sigma(N) - (\mathcal{I}_\infty - \mathcal{P})^{-1}\,\mathcal{C}\,(\mathcal{I}_\infty - \mathcal{P})^{-1}\right) = (\mathcal{I}_\infty - \mathcal{P})^{-1}\,\mathcal{Q}_\infty\,(\mathcal{I}_\infty - \mathcal{P})^{-1} \,,$$

which completes the proof. □

Corollary 16. *Under the assumptions of Proposition 15, for any matrix norm $\|\cdot\|$ there exists a constant $C > 0$ and $N_0 \in \mathbb{N}$ such that*

$$\|\Sigma(N) - (\mathcal{I}_\infty - \mathcal{P})^{-1}\,\mathcal{C}\,(\mathcal{I}_\infty - \mathcal{P})^{-1}\| \le \frac{C}{N} \quad \textit{for all } N \ge N_0.$$

In other words, for large time-batch size N, the covariance matrix $\Sigma(N)$ decays at rate $1/N$ as N increases.

Proof. For clarity, let

$$f(N) = \Sigma(N) - (\mathcal{I}_\infty - \mathcal{P})^{-1}\,\mathcal{C}\,(\mathcal{I}_\infty - \mathcal{P})^{-1} \,.$$

By Proposition 15 we have

$$\lim_{N\to\infty} N f(N) = (\mathcal{I}_\infty - \mathcal{P})^{-1}\,\mathcal{Q}_\infty\,(\mathcal{I}_\infty - \mathcal{P})^{-1} := \Sigma_\infty.$$

Let $\|\cdot\|$ be any matrix norm. The map $A \mapsto \|A\|$ is continuous, so

$$\lim_{N\to\infty} \left\| N\Sigma(N) - N\,(\mathcal{I}_\infty - \mathcal{P})^{-1}\,\mathcal{C}\,(\mathcal{I}_\infty - \mathcal{P})^{-1} - \Sigma_\infty \right\| = 0.$$

Hence there exists $N_0 \in \mathbb{N}$ such that, for all $N \ge N_0$,

$$\|N f(N) - \Sigma_\infty\| \le 1.$$

For such N we obtain

$$\|N f(N)\| \le \|N f(N) - \Sigma_\infty\| + \|\Sigma_\infty\| \le 1 + \|\Sigma_\infty\| \implies \|f(N)\| \le \frac{1 + \|\Sigma_\infty\|}{N} \quad \text{for all } N \ge N_0.$$

Setting $C := 1 + \|\Sigma_\infty\|$ completes the proof. □

E.4 Proof of proposition 5

For clarity of the proof, we first present two lemmata as important stepping stones.

E.4.1 LEMMATA

Lemma 17 (Kolmogorov semigroup expansion)**.** *Let X_t solves the Itô SDE $\mathrm{d}X_t = f(t, X_t)\mathrm{d}t + g(t, X_t)\mathrm{d}W_t$ and $\phi \in \mathbb{C}_b^4$. Then with a small time increment δ,*

$$\mathbb{E}[\phi(X_{t+\delta})|X_t = x] = \phi(x) + \delta\mathcal{L}_t\phi(x) + \frac{\delta^2}{2}\mathcal{L}_t^2\phi(x) + \mathcal{O}(\delta^2) \,, \tag{69}$$

or equivalently

$$\mathbb{E}_{p_{t+\delta}}[\phi(X)] = \mathbb{E}_{p_t}[\phi(X)] + \delta\mathbb{E}_{p_t}[\mathcal{L}_t\phi(X)] + \frac{\delta^2}{2}\mathbb{E}_{p_t}[\mathcal{L}_t^2\phi(X)] + \mathcal{O}(\delta^2) \,, \tag{70}$$

where $\mathcal{L}_t\phi := f(t,\cdot)^\top\nabla\phi + \frac{1}{2}g(t,\cdot)g(t,\cdot)^\top\Delta\phi$ is the time-dependent backward generator.

Lemma 18 (Generator-Adjoint Identity)**.** *Given the Itô SDE, ϕ, and $\mathcal{L}_t$ defined in lemma 17, we have*

$$\mathbb{E}_{p_t}[\mathcal{L}_t\phi(X)] = \mathbb{E}_{p_t}\left[\phi(X)\frac{\partial}{\partial t}\log p_t(X)\right] \,. \tag{71}$$

E.4.2 PROOF OF PROPOSITION

Proposition 19. *Let $t \in [0, T)$ and $\delta > 0$, we have*

$$\lim_{\delta\to 0^+}\frac{8}{\delta^2}\left(\mathcal{L}_{\mathrm{clf}}(\theta; t, t+\delta) - \log 2\right) = \mathcal{L}_{tSM}(\theta; t) + C \,, \tag{72}$$

where $C = \mathbb{E}_{p_t}\left[\left(\frac{\partial}{\partial t}\log p_t(Y_t)\right)^2\right]$ is a constant with respect to θ.

Proof. In the binary case, we first rewrite the classification loss as follows

$$\begin{aligned}
\mathcal{L}_{\mathrm{clf}}(\theta; t, t+\delta) &= -\frac{1}{2}\left\{\mathbb{E}_{p_t}\left[\log\frac{p_t^\theta(x)}{p_t^\theta(x) + p_{t+\delta}^\theta(x)}\right] + \mathbb{E}_{p_{t+\delta}}\left[\log\frac{p_{t+\delta}^\theta(x)}{p_t^\theta(x) + p_{t+\delta}^\theta(x)}\right]\right\} \\
&= -\frac{1}{2}\left\{\underbrace{\mathbb{E}_{p_t}\left[\log\sigma(\log p_t^\theta(x) - \log p_{t+\delta}^\theta(x))\right]}_{A} + \underbrace{\mathbb{E}_{p_{t+\delta}}\left[\log\sigma(\log p_{t+\delta}^\theta(x) - \log p_t^\theta(x))\right]}_{B}\right\},
\end{aligned}$$

where $\sigma(z) = 1/(1 + e^{-z})$ is the sigmoid function. By Taylor expansion,

$$\begin{aligned}
\log p_t^\theta(x) - \log p_{t+\delta}^\theta(x) &= -\delta\frac{\partial}{\partial t}\log p_t^\theta(x) + \mathcal{O}(\delta^2), \\
\log\sigma(z) &= -\log 2 + \frac{z}{2} - \frac{z^2}{8} + \mathcal{O}(z^4).
\end{aligned}$$

Therefore, A and B can be simplified as

$$\begin{aligned}
A &= -\log 2 + \mathbb{E}_{p_t}\left[-\frac{\delta}{2}\frac{\partial}{\partial t}\log p_t^\theta(x) - \frac{\delta^2}{8}\left(\frac{\partial}{\partial t}\log p_t^\theta(x)\right)^2\right] + \mathcal{O}(\delta^3), \\
B &= -\log 2 + \mathbb{E}_{p_{t+\delta}}\left[\frac{\delta}{2}\frac{\partial}{\partial t}\log p_t^\theta(x) - \frac{\delta^2}{8}\left(\frac{\partial}{\partial t}\log p_t^\theta(x)\right)^2\right] + \mathcal{O}(\delta^3).
\end{aligned}$$

Next, we use Lemma 17 with $\phi_1(x) = \frac{\partial}{\partial t}\log p_t^\theta(x)$ and $\phi_2(x) = \left(\frac{\partial}{\partial t}\log p_t^\theta(x)\right)^2$ as follows

$$\begin{aligned}
\mathbb{E}_{p_{t+\delta}}\left[\phi_1(x)\right] &= \mathbb{E}_{p_t}\left[\frac{\partial}{\partial t}\log p_t^\theta(x)\right] + \delta\mathbb{E}_{p_t}\left[\mathcal{L}_t\frac{\partial}{\partial t}\log p_t^\theta(x)\right] + \mathcal{O}(\delta^2), \\
\mathbb{E}_{p_{t+\delta}}\left[\phi_2(x)\right] &= \mathbb{E}_{p_t}\left[\left(\frac{\partial}{\partial t}\log p_t^\theta(x)\right)^2\right] + \mathcal{O}(\delta).
\end{aligned}$$

Plugging $\mathbb{E}_{p_{t+\delta}}[\phi_1(x)]$ and $\mathbb{E}_{p_{t+\delta}}[\phi_2(x)]$, B can be written as

$$B = -\log 2 + \frac{\delta}{2}\mathbb{E}_{p_t}\left[\frac{\partial}{\partial t}\log p_t^\theta(x)\right] + \frac{\delta^2}{2}\mathbb{E}_{p_t}\left[\mathcal{L}_t\frac{\partial}{\partial t}\log p_t^\theta(x)\right] - \frac{\delta^2}{8}\mathbb{E}_{p_t}\left[\left(\frac{\partial}{\partial t}\log p_t^\theta(x)\right)^2\right] + \mathcal{O}(\delta^3). \quad (73)$$

Therefore,

$$A + B = -2\log 2 + \frac{\delta^2}{2}\mathbb{E}_{p_t}\left[\mathcal{L}_t\frac{\partial}{\partial t}\log p_t^\theta(x)\right] - \frac{\delta^2}{4}\mathbb{E}_{p_t}\left[\left(\frac{\partial}{\partial t}\log p_t^\theta(x)\right)^2\right] + \mathcal{O}(\delta^3).$$

By lemma 18, the $\mathbb{E}_{p_t}\left[\mathcal{L}_t\frac{\partial}{\partial t}\log p_t^\theta(x)\right]$ term could be written as

$$\mathbb{E}_{p_t}\left[\mathcal{L}_t\frac{\partial}{\partial t}\log p_t^\theta(x)\right] = \mathbb{E}_{p_t}\left[\frac{\partial}{\partial t}\log p_t^\theta(x)\frac{\partial}{\partial t}\log p_t(x)\right], \quad (74)$$

which results in

$$\begin{aligned} &A + B + 2\log 2 \\ =& -\frac{\delta^2}{4}\left(\mathbb{E}_{p_t}\left[\left(\frac{\partial}{\partial t}\log p_t^\theta(x)\right)^2\right] - 2\mathbb{E}_{p_t}\left[\frac{\partial}{\partial t}\log p_t^\theta(x)\frac{\partial}{\partial t}\log p_t(x)\right]\right) + \mathcal{O}(\delta^3) \\ =& -\frac{\delta^2}{4}\left(\mathbb{E}_{p_t}\left[\left(\frac{\partial}{\partial t}\log p_t^\theta(x) - \frac{\partial}{\partial t}\log p_t(x)\right)^2\right] + \mathbb{E}_{p_t}\left[\left(\frac{\partial}{\partial t}\log p_t(x)\right)^2\right]\right) + \mathcal{O}(\delta^3). \end{aligned}$$

Therefore, given a small time-increment δ, the binary-classification loss can be written as

$$\mathcal{L}_{\text{clf}}(\theta; t, t+\delta) = \frac{\delta^2}{8}\mathbb{E}_{p_t}\left[\left(\frac{\partial}{\partial t}\log p_t^\theta(x) - \frac{\partial}{\partial t}\log p_t(x)\right)^2\right] + \frac{\delta^2}{8}\mathbb{E}_{p_t}\left[\left(\frac{\partial}{\partial t}\log p_t(x)\right)^2\right] + \log 2 + \mathcal{O}(\delta^3). \quad (75)$$

Since $C = \frac{\delta^2}{8}\mathbb{E}_{p_t}\left[\left(\frac{\partial}{\partial t}\log p_t(x)\right)^2\right] + \log 2$ is constant *w.r.t.* θ, it recovers the tSM regularization. □

E.5 Proof of proposition 6

Proposition 6. *Let $\delta > 0$. In the small step-size regime, the Bayes objective $\mathcal{L}_{\text{Bayes}}$ recovers the Fokker–Planck regularization $\mathcal{L}_{\text{FPE}}$, i.e.,*

$$\lim_{\delta\to 0}\frac{1}{\delta}\mathcal{L}_{\text{Bayes}}(\theta; t, t+\delta) = \mathcal{L}_{\text{FPE}}(\theta; t)\ . \quad (76)$$

Proof. From Equation (35), we write the full learning objective of the FPE regularzation for reference:

$$\mathcal{L}_{\text{FPE}}(\theta) = \int_0^1 \mathbb{E}_{p_t}\left[\Big(\partial_t \log p_t^\theta(y) + \nabla\cdot\alpha_t(y) + \alpha_t(y)^\top\nabla\log p_t^\theta(y) - \frac{1}{2}\beta_t^2\nabla\cdot\nabla\log p_t^\theta(y) - \frac{1}{2}\beta_t^2\|\nabla\log p_t^\theta(y)\|^2\Big)^2\right]\mathrm{d}t\ . \quad (77)$$

Now, let's look at the Bayes regularization. Given the EM discretizations for both the forward (38) and backward (39) kernels and assume that the $(y_{t-\delta}, y_t)$ are sampled forwardly, *i.e.* $y_t =$

$y_{t-\delta} + \alpha_{t-\delta}(y_{t-\delta})\delta + \beta_{t-\delta}\sqrt{\delta}z$, where $z \sim \mathcal{N}(0, I)$ and $\delta > 0$. In this case, the forward kernel should be corrected for the volume change induced by the drift α, *i.e.*

$$p_{t|t-\delta}(y_t|y_{t-\delta}) \approx \frac{1}{1 + \delta\nabla \cdot \alpha_{t-\delta}(y_{t-\delta})}\mathcal{N}(y_t; y_{t-\delta} + \alpha_{t-\delta}(y_{t-\delta})\delta, \beta_{t-\delta}^2\delta I). \tag{78}$$

Let $b_t(y) = \alpha_t(y) - \beta_t^2\nabla \log p_t(y)$, then the log forward and backward kernels can be further approximated using Taylor expansion around (t, y_t) as follows

$$\begin{aligned}
\log p_{t|t-\delta}(y_t|y_{t-\delta}) &\approx C - \frac{1}{2}\|z\|^2 - \log(1 + \delta\nabla \cdot \alpha_{t-\delta}(y_{t-\delta})) \\
&\approx C - \frac{1}{2}\|z\|^2 - \delta\nabla \cdot \alpha_t(y) + \mathcal{O}(\delta)\ , \qquad (79) \\
\log p_{t-\delta|t}(y_{t-\delta}|x_t) &\approx C - \frac{1}{2\beta_t^2\delta}\|\alpha_{t-\delta}(y_{t-\delta})\delta + \beta_{t-\delta}\sqrt{\delta}z - b_t(y_t)\delta\|^2 \\
&= C - \frac{1}{2\beta_t^2\delta}\|\alpha_t(y_t)\delta + \beta_t\sqrt{\delta}z - b_t(y_t)\delta + \mathcal{O}(\delta)\|^2 \\
&= C - \frac{1}{2}\|z\|^2 - \frac{\delta}{2\beta_t^2}\|\alpha_t(y_t) - b_t(y_t)\|^2 - \frac{\sqrt{\delta}}{\beta_t}z^\top(\alpha_t(y_t) - b_t(y_t)) + \mathcal{O}(\delta)\ , \qquad (80)
\end{aligned}$$

where $C = -\frac{d}{2}\log 2\pi - d\log g(t)$. Plugging back $b_t(y) = \alpha_t(y) - \beta_t^2\nabla \log p_t(y)$, the log Radon–Nikodym derivative can be approximated as

$$\log \frac{p_{t|t-\delta}(y_t|y_{t-\delta})}{p_{t-\delta|t}(y_{t-\delta}|x_t)} \approx \frac{\beta_t^2\delta}{2}\|\nabla \log p_t(y_t)\|^2 + \beta_t\sqrt{\delta}z^\top\nabla \log p_t(y_t) - \delta\nabla \cdot \alpha_t(y_t) + \mathcal{O}(\delta)\ . \tag{81}$$

In meanwhile, one could Taylor expand the log marginal density $\log p_{t-\delta}(y_{t-\delta})$ around (t, y_t)

$$\begin{aligned}
\log p_{t-\delta}(y_{t-\delta}) \approx \log p_t(y_t) - \frac{\partial}{\partial t}\log p_t(y_t)\delta + \nabla \log p_t(y_t)^\top(y_{t-\delta} - y_t) \\
+ \frac{1}{2}(y_{t-\delta} - y_t)^\top\nabla^2 \log p_t(y_t)(y_{t-\delta} - y_t) + \mathcal{O}(\delta^{3/2})\ . \qquad (82)
\end{aligned}$$

By plugging $y_t - y_{t-\delta} \approx \alpha_t(y_t)\delta + \beta_t\sqrt{\delta}z + \mathcal{O}(\delta^{3/2})$, we have

$$\begin{aligned}
\log p_{t-\delta}(y_{t-\delta}) \approx \log p_t(y_t) - \frac{\partial}{\partial t}\log p_t(y_t)\delta - \nabla \log p_t(y_t)^\top\alpha_t(y_t)\delta \\
- \beta_t\sqrt{\delta}z^\top\nabla \log p_t(y_t) + \frac{\beta_t^2\delta}{2}z^\top\nabla^2 \log p_t(y_t)z + \mathcal{O}(\delta^{3/2})\ . \qquad (83)
\end{aligned}$$

Now, we could plug all the approximations together, and take the expectation

$$\begin{aligned}
&\frac{1}{\delta}\mathbb{E}_z\left[\log p_{t-\delta}(y_{t-\delta}) - \log p_t(y_t) + \log \frac{p_{t|t-\delta}(y_t|y_{t-\delta})}{p_{t-\delta|t}(y_{t-\delta}|x_t)}\right] \\
&= \left(-\frac{\partial}{\partial t}\log p_t(y_t) - \nabla \cdot \alpha_t(y_t) - \nabla \log p_t(y_t)^\top f(t, x_t) + \frac{\beta_t^2}{2}(\mathrm{tr}(\nabla^2 \log p_t(y_t)) + \|\nabla \log p_t(y_t)\|^2)\right)\ , \qquad (84)
\end{aligned}$$

which recovers the Fokker Plank residual when $\delta \to \infty$, and therefore RNE regularization recovers the Fokker Plank regularization in the limit. □

F Extend DiffCLF with Bregman Divergence

The effective training objective of `DiffCLF` is to estimate the density ratios between noise levels, i.e. p_t^θ/p_s^θ for $s, t \in [0, T]$. Bregman divergence (Bregman, 1967; Gutmann & ichiro Hirayama, 2012) could be utilized to generalize this density-ratio estimation framework.

F.1 BREGMAN DIVERGENCE

The Bregman divergence (Bregman, 1967) between two functionals $f : \mathbb{R}^d \to \mathbb{R}^m$ and $g : \mathbb{R}^d \to \mathbb{R}^m$ within an underlying measure μ is defined as

$$D_\phi^\mu(f, g) = \mathbb{E}_\mu \left[\phi(f) - \phi(g) - \nabla\phi(g) \cdot (f - g)\right] , \tag{85}$$

where $\phi : \mathbb{R}^m \to \mathbb{R}$ is a strighlty convex *generator* and $\nabla\phi(g)$ refers to $\nabla_g\phi(g)$. In the following context, we stick using f as some known quantities while g is the learnable components. Therefore, $\mathbb{E}_\mu\phi(f)$ is a constant with respect to the learnable parameters, and hence we could equivalently minimize the following loss,

$$\mathcal{L}_\phi^\mu(f, g) = \mathbb{E}_\mu \left[-\phi(g) + \nabla\phi(g) \cdot g\right] - \mathbb{E}_\mu \left[\nabla\phi(g) \cdot f\right] . \tag{86}$$

F.2 BINARY CASE

In the binary case, `DiffCLF` aims to match the p_t^θ/p_s^θ with the ground truth density ratio p_t/p_s, for any $t, s \in [0, T]$. That is,

$$f = \frac{p_t}{p_s} \quad \& \quad g = \frac{p_t^\theta}{p_s^\theta} := r_{st}^\theta .$$

Let $\mu = p_s$,

$$\begin{aligned}
\mathcal{L}_\phi^\mu(f, g) &= \mathbb{E}_{p_s} \left[-\phi\left(r_{st}^\theta(x_s)\right) + \phi'\left(r_{st}^\theta(x_s)\right) r_{st}^\theta(x_s)\right] - \mathbb{E}_{p_s} \left[\phi'\left(r_{st}^\theta(x_s)\right) \frac{p_t(x_s)}{p_s(x_s)}\right] \\
&= \mathbb{E}_{p_s} \left[-\phi\left(r_{st}^\theta(x_s)\right) + \phi'\left(r_{st}^\theta(x_s)\right) r_{st}^\theta(x_s)\right] - \mathbb{E}_{p_t} \left[\phi'\left(r_{st}^\theta(x_t)\right)\right] ,
\end{aligned} \tag{87}$$

where we use ϕ' to denote the derivative of ϕ for clarity as ϕ is now a scalar function.

Let $\phi(r) = r \log r - (1 + r) \log(1 + r)$, where $\phi'(r) = \log r - \log(1 + r)$, the above loss recovers the binary `DiffCLF` loss (Equation (12)):

$$\begin{aligned}
\mathbb{E}_{p_s} \Big[- \Big(& \cancel{r_{st}^\theta(x_s) \log r_{st}^\theta(x_s)} - (1 + \cancel{r_{st}^\theta(x_s)}) \log(1 + r_{st}^\theta(x_s)) \Big) \\
& + \cancel{r_{st}^\theta(x_s) \log \frac{r_{st}^\theta(x_s)}{(1 + r_{st}^\theta(x_s))}} \Big] - \mathbb{E}_{p_t} \left[\log \frac{r_{st}^\theta(x_s)}{(1 + r_{st}^\theta(x_s))}\right] ,
\end{aligned}$$

which can be simplified as

$$\mathbb{E}_{p_s} \left[\log\left(1 + r_{st}^\theta(x_s)\right)\right] - \mathbb{E}_{p_t} \left[\log \frac{r_{st}^\theta(x_t)}{1 + r_{st}^\theta(x_t)}\right] \tag{88}$$

$$= - \mathbb{E}_{p_s} \left[\log \left(\frac{p_s^\theta(x_s)}{p_s^\theta(x_s) + p_t^\theta(x_s)}\right)\right] - \mathbb{E}_{p_t} \left[\log \frac{p_t^\theta(x_t)}{p_s^\theta(x_t) + p_t^\theta(x_t)}\right] \tag{89}$$

$$= 2\mathcal{L}_{\text{clf}}(\theta; s, t) . \tag{90}$$

Another possible choices of $\phi(r)$ include $r \log r - r$, $-\log r$, and $\frac{1}{2}r^2$. However, these alternative choices arise singularity on either $r = 0$ or $r = \infty$. Therefore, we argue that $\phi(r) = r \log r - (1 + r) \log(1 + r)$ is the optimal option.

F.3 MULTICLASS CASE

In the multiclass case, `DiffCLF` aims to match all the density ratios $p_{t_i}^\theta/p_{t_j}^\theta$, given $i, j \in [\![1, N]\!]$. For clarity, we write p_i as p_{t_i}. The functionals f and g are vector-valued, defined as:

$$f = \left[\frac{p_1}{\sum_{j=1}^N p_j}, ..., \frac{p_N}{\sum_{j=1}^N p_j}\right] \quad \text{and} \quad g = \left[\frac{p_1^\theta}{\sum_{j=1}^N p_j^\theta}, ..., \frac{p_N^\theta}{\sum_{j=1}^N p_j^\theta}\right] := r_{1:N}^\theta ,$$

where $r_j^\theta = p_j^\theta / \sum_i p_i^\theta$. Let μ be the simple mixture of $\{p_j\}_{j=1}^N$, i.e. $\mu = \bar{p} := \frac{1}{N}\sum_{j=1}^N p_j$,

$$\begin{aligned}\mathcal{L}_\phi^\mu(f,g) &= \mathbb{E}_{\bar{p}}\left[-\phi\left(r_{1:N}^\theta(x)\right) + \nabla\phi\left(r_{1:N}^\theta(x)\right)\cdot r_{1:N}^\theta(x)\right] - \mathbb{E}_{\bar{p}}\left[\nabla\phi\left(r_{1:N}^\theta(x)\right)\cdot\frac{\mathbf{p}(x)}{\sum_{j=1}^N p_j(x)}\right] \\ &= \mathbb{E}_{\bar{p}}\left[-\phi\left(r_{1:N}^\theta(x)\right) + \nabla\phi\left(r_{1:N}^\theta(x)\right)\cdot r_{1:N}^\theta(x)\right] - \frac{1}{N}\sum_{j=1}^N \mathbb{E}_{p_j}\left[\left(\nabla\phi\left(r_{1:N}^\theta(x)\right)\right)_j\right] ,\end{aligned} \tag{91}$$

where $\mathbf{p} = [p_1, ..., p_N]$.

Let $\phi(r) = \sum_i r_i \log r_i$, where $\nabla\phi(r) = [1+\log r_1, ..., 1+\log r_N]$, then

$$\begin{aligned}\mathcal{L}_\phi^\mu(f,g) &= \mathbb{E}_{\bar{p}}\left[\cancel{-\sum_{j=1}^N r_j^\theta(x)\log r_j^\theta(x)} + \sum_{j=1}^N (1+\cancel{\log r_j^\theta(x)})r_j^\theta(x)\right] - \frac{1}{N}\sum_{j=1}^N \mathbb{E}_{p_j}\left[1+\log r_j^\theta(x_j)\right] \\ &= \mathbb{E}_{\bar{p}}\left[\sum_{j=1}^N r_j^\theta(x)\right] - 1 - \frac{1}{N}\sum_{j=1}^N \mathbb{E}_{p_j}\left[\log r_j^\theta(x_j)\right] \\ &\overset{(i)}{=} -\frac{1}{N}\sum_{j=1}^N \mathbb{E}_{p_j}\left[\log\frac{p_j^\theta(x_j)}{\sum_i p_i^\theta(x_j)}\right] ,\end{aligned} \tag{92}$$

recovers the multiclass `DiffCLF` loss defined in Equation (10), where (i) is based on $\sum_i r_i^\theta \equiv 1$.

Another possible choices of $\phi(r)$ include $\frac{1}{2}\sum_i r_i^2$ and $\sum_i 1/r_i$. In particular, one could also choose $\phi(r) = \sum_i r_i \log r_i - (1+r_i)\log(1+r_i)$ as the binary case, which leads to

$$\begin{aligned}\mathcal{L}_\phi^\mu(f,g) &= \mathbb{E}_{\bar{p}}\Bigg[-\sum_{j=1}^N \left(\cancel{r_j^\theta(x)\log r_j^\theta(x)} - (1+\cancel{r_j^\theta(x)})\log(1+r_j^\theta(x))\right) \\ &\qquad + \cancel{\sum_{j=1}^N r_j^\theta(\log r_j^\theta(x) - \log(1+r_j^\theta(x)))}\Bigg] - \frac{1}{N}\sum_{j=1}^N \mathbb{E}_{p_j}\left[\log r_j^\theta(x_j) - \log(1+r_j^\theta(x_j))\right] \\ &= \frac{1}{N}\sum_{j=1}^N \mathbb{E}_{p_j}\left[2\log(1+r_j^\theta(x_j)) - \log r_j^\theta(x_j)\right] \\ &= \frac{1}{N}\sum_{j=1}^N \mathbb{E}_{p_j}\left[2\log\left(1+\frac{p_j^\theta(x_j)}{\sum_i p_i^\theta(x_j)}\right) - \log\frac{p_j^\theta(x_j)}{\sum_i p_i^\theta(x_j)}\right] .\end{aligned} \tag{93}$$

G Extension to Discrete Diffusions

Recent works Campbell et al. (2022); Meng et al. (2023); Lou et al. (2023); Campbell et al. (2024); Gat et al. (2024); Shaul et al. (2025) extend diffusion models to discrete state spaces $\mathbb{Y}$ by formulating the forward noising process as a continuous-time Markov chain (CTMC). A CTMC is specified by a family of (possibly time-dependent) transition rate matrices $(Q_t)_t$, a.k.a. *generators*, where for each t: (i) $Q_t = (Q_t(y,y'))_{y,y'\in\mathbb{Y}}$, (ii) $Q_t(y,y) = -\sum_{y'\neq y} Q_t(y,y')$ for any $y \in \mathbb{Y}$, and (iii) $Q_t(y,y') \geq 0$ for all $y \neq y' \in \mathbb{Y}$. We write

$$Y_t \sim \mathrm{CTMC}(Q_t) ,$$

to mean that $Y_0 \sim p_0$ followed by infinitesimal transitions governed by Q_t. As a stochastic process starting from a target distribution p_0, $(Q_t)_t$ induces a sequence of intermediate marginals $(p_t)_t$, which satisfy the *Kolmogorov forward equation* (also called the master equation):

$$\frac{\partial}{\partial t}p_t(y) = \sum_{y'} p_t(y')\,Q_t(y',y) \iff \partial_t p_t = Q_t^\top p_t . \tag{94}$$

Analogous to the Stein score in continuous spaces, one can define the *concrete score* in discrete spaces as the vector of marginal density ratios

$$S_t(y)_{y'} := \frac{p_t(y')}{p_t(y)} \ .$$

In terms of this score, the log-density dynamics admit the compact form

$$\frac{\partial}{\partial t} \log p_t(y) = \frac{1}{p_t(y)} \sum_{y'} p_t(y') \, Q_t(y', y) = \sum_{y'} S_t(y)_{y'} \, Q_t(y', y) \ . \tag{95}$$

Finally, under mild regularity conditions, the *time-reversed process* $(Y_t)_t$ is again a CTMC with generator $(\tilde{Q}_t)_t$ (Kelly, 2011) given by

$$Y_t \sim \mathrm{CTMC}(\tilde{Q}_t), \quad \text{where} \quad \tilde{Q}_t(y, y') = \begin{cases} Q_t(y', y) \, \frac{p_t(y')}{p_t(y)}, & y \neq y' \ , \\ -\sum_{y' \neq y} \tilde{Q}_t(y, y'), & y = y' \ . \end{cases} \tag{96}$$

Though the marginals $(p_t)_t$ are intractable, one could train time-dependent neural networks $s_t^\theta(y)$ to minimize the following *Score Entropy* (SE) loss

$$\mathcal{L}_{\mathrm{SE}}(\theta; t) = \mathbb{E}_{p_t} \left[\sum_{y \neq Y_t \in \mathbb{Y}} Q_t(Y_t, y) \left(S_t(Y_t)_y \log \frac{S_t(Y_t)_y}{s_t^\theta(Y_t)_y} - S_t(Y_t)_y + s_t^\theta(Y_t)_y \right) \right], \tag{97}$$

(98)

by optimizing the following *Conditional Score Entropy* loss analogous to DSM

$$\mathcal{L}_{\mathrm{CSE}}(\theta; t) = \mathbb{E}_{p_0} \mathbb{E}_{p_{t|0}} \left[\sum_{y \neq Y_t \in \mathbb{Y}} Q_t(Y_t, y) \left(-S_t(Y_t|Y_0)_y \log s_t^\theta(Y_t)_y + s_t^\theta(Y_t)_y \right) \right], \tag{99}$$

where $S_t(Y_t|Y_0)$ is the *conditional concrete score*, $\nabla_\theta \mathcal{L}_{\mathrm{CSE}}(\theta; t) = \nabla_\theta \mathcal{L}_{\mathrm{SE}}(\theta; t)$, and $p_{t|0}$ is the conditional distribution obtained by solving the following ODE where $S_t(Y_t|Y_0)$ is the *conditional concrete score*, $\nabla_\theta \mathcal{L}_{\mathrm{CSE}}(\theta; t) = \nabla_\theta \mathcal{L}_{\mathrm{SE}}(\theta; t)$, and $p_{t|0}$ is the conditional distribution obtained by solving the following ODE

$$\frac{\partial}{\partial t} p_{t|0}(y|y_0) = \sum_{y'} p_{t|0}(y'|y_0) Q_t(y', y) \quad \text{with } p_{0|0}(y|y_0) = \delta_{y_0}(y). \tag{100}$$

Energy-based training. Similar to eq. (2), one could define a family of EBMs on $\mathbb{Y}$ as follows

$$p_t^\theta(y_t) = \exp(-\mathrm{U}_t^\theta(y_t))/\mathcal{Z}_t^\theta, \quad \mathcal{Z}_t^\theta = \exp(\mathrm{F}_t^\theta) = \sum_{y_t \in \mathbb{Y}} \exp(-\mathrm{U}_t^\theta(y_t)) \ . \tag{101}$$

Simply plugging p_t^θ into eq. (99) we have

$$\mathcal{L}_{\mathrm{CSE}}(\theta; t) = \mathbb{E}_{p_0} \mathbb{E}_{p_{t|0}} \left[\sum_{y \neq Y_t \in \mathbb{Y}} Q_t(Y_t, y) \left(-S_t(Y_t)_y \log \frac{p_t^\theta(y)}{p_t^\theta(Y_t)} + \frac{p_t^\theta(y)}{p_t^\theta(Y_t)} \right) \right] \tag{102}$$

$$= \mathbb{E}_{p_0} \mathbb{E}_{p_{t|0}} \left[\sum_{y \neq Y_t \in \mathbb{Y}} Q_t(Y_t, y) \left\{ S_t(Y_t)_y (\mathrm{U}_t^\theta(y) - \mathrm{U}_t^\theta(Y_t)) + \exp(\mathrm{U}_t^\theta(Y_t) - \mathrm{U}_t^\theta(y)) \right\} \right] \tag{103}$$

Combined with classification loss. Therefore, it is straightforward to combine the classification loss 10 with Equation (99) to train an energy-based discrete Diffusion model.

H EXPERIMENTAL DETAILS

H.1 GAUSSIAN MIXTURES AND CLOSED FORM EXPRESSIONS FOR DMS AND SIS

Mixture of Gaussians (MOG) is distribution having the following density

$$\pi(x) = \sum_{n=1}^{N} w_n \mathcal{N}(x; \mu_n, \Sigma_n) \ .$$

Diffusion Models case. In DMs, we require the exact marginal density, which is a convolution of the noising kernel and the target distribution. Assume the noising kernel is $p_{t|0}(y_t|y_0) = \mathcal{N}(y_t; S(t)y_0, \gamma(t)^2 \mathrm{I}_d)$, we have

$$p_t(y_t) = \int p_0(y_0) p_{t|0}(y_t|y_0) dy_0 = \sum_{n=1}^{N} w_n \mathcal{N}(y_t; S(t)\mu_n, S(t)^2\Sigma_n + \gamma(t)^2 \mathrm{I}_d) \ , \tag{104}$$

again a MOG. Therefore the marginal density and score are tractable.

Stochastic Interpolant case. In SIs, we consider $p_0(y) = \sum_{n=1}^{N} w_n \mathcal{N}(y; \mu_n, \Sigma_n)$ and $p_1(y) = \sum_{m=1}^{M} \tilde{w}_m \mathcal{N}(y; \tilde{\mu}_m, \Sigma_m)$ are both MOGs with independent coupling and a linear interpolation $I_t(y_0, y_1) = (1-t)y_0 + ty_1$, therefore

$$p_t(y_t) = \iint p_0(y_0) p_1(y_1) \mathcal{N}(y_t; I_t(y_0, y_1), \gamma(t)^2 \mathrm{I}_d) dy_0 dy_1 \tag{105}$$

$$= \sum_n \sum_m w_n \tilde{w}_m \mathcal{N}(y_t; (1-t)\mu_n + t\tilde{\mu}_m, (1-t)^2\Sigma_n + t^2\tilde{\Sigma}_m + \gamma(t)^2 \mathrm{I}_d) \ , \tag{106}$$

allowing exact marginal density and score calculation. Moreover, we require the velocity in SIs, which is $\mathbb{E}[\partial_t I_t(Y_0, Y_1)|Y_t = y_t] = \mathbb{E}[Y_1 - Y_0|Y_t = y_t]$. To get the analytical velocity, notice that

$$p(y_0, y_1 \mid y_t) = \sum_{n=1}^{N} \sum_{m=1}^{M} \pi_{n,m}(y_t) \, \mathcal{N}\left(\begin{bmatrix} y_0 \\ y_1 \end{bmatrix}; \begin{bmatrix} \mu_{0|t}^{(n,m)} \\ \mu_{1|t}^{(n,m)} \end{bmatrix}, \ \Sigma_{|t}^{(n,m)} \right) \ , \tag{107}$$

$$\pi_{n,m}(y_t) = \frac{w_n \tilde{w}_m \, \mathcal{N}\left(y_t; \bar{\mu}_{n,m}, S_{n,m}\right)}{\sum_{n'=1}^{N} \sum_{m'=1}^{M} w_{n'} \tilde{w}_{m'} \, \mathcal{N}\left(y_t; \bar{\mu}_{n',m'}, S_{n',m'}\right)}, \tag{108}$$

with

$$\bar{\mu}_{n,m} = (1-t)\mu_n + t\tilde{\mu}_m \ , \tag{109}$$

$$S_{n,m} = (1-t)^2\Sigma_n + t^2\tilde{\Sigma}_m + \gamma(t)^2 I_d \ , \tag{110}$$

$$\mu_{0|t}^{(n,m)} = \mu_n + (1-t)\Sigma_n S_{n,m}^{-1}(y_t - \bar{\mu}_{n,m}) \ , \tag{111}$$

$$\mu_{1|t}^{(n,m)} = \tilde{\mu}_m + t\tilde{\Sigma}_m S_{n,m}^{-1}(y_t - \bar{\mu}_{n,m}) \ . \tag{112}$$

Therefore, the exact velocity in this case is given by

$$v_t(y_t) = \mathbb{E}[Y_1 - Y_0 \mid Y_t = y_t] = \sum_{n=1}^{N} \sum_{m=1}^{M} \pi_{n,m}(y_t) \ \left(\mu_{1|t}^{(n,m)} - \mu_{0|t}^{(n,m)} \right) \ . \tag{113}$$

H.2 ANALYTICAL COMPARISON WITH DSM ON MOG

In this section, we provide the experimental setup for Gaussian mixture experiments along with additional results.

H.2.1 GAUSSIAN MIXTURE DESIGN

We study two types of Gaussian mixtures. The first, introduced in Midgley et al. (2023), is a widely used benchmark consisting of 40 Gaussians with uniform weights (MOG-40). The means are sampled from $\mathrm{U}([-40, 40]^d)$, and all components share the same covariance $\log(1+e)\mathrm{I}_d$. The second, taken from Grenioux et al. (2024); Noble et al. (2025), is a two-component mixture with modes at $-5 \times \mathbf{1}_d$ and $+5 \times \mathbf{1}_d$ (where $\mathbf{1}_d$ denotes the d-dimensional vector of ones), covariance $5 \times 10^{-2}, \mathrm{I}_d$, and imbalanced weights $2/3$ and $1/3$. For training, we standardize these distributions (subtracting the mean and dividing by the standard deviation).

H.2.2 ARCHITECTURE, TRAINING AND EVALUATION DETAILS

We train log-densities using three settings: (i) $\mathcal{L}_{\text{DSM}}$ alone, (ii) a convex combination of $\mathcal{L}_{\text{DSM}}$ with either $\mathcal{L}_{\text{clf}}$ or $\mathcal{L}_{\text{C}t\text{SM}}$. In the latter case, simply summing the two losses generally worked best.

Diffusion Model. For DMs, we adopt the Variance Preserving (VP) schedule Song et al. (2021b) with a linear β-schedule ending at $\beta_{\max} = 20$. Time is discretized linearly into 512 steps between 10^{-4} and $1 - 10^{-4}$. We follow the energy parameterization of Thornton et al. (2025), use the DSM weighting from Karras et al. (2022), and implement a 4-layer MLP of width 128 with sinusoidal time embeddings Song et al. (2021b). The conditional t-SM loss is reweighted by $\gamma^2/\dot{\gamma}^2$ as recommended by Yu et al. (2025). Models are trained on 60k samples for 500 epochs with DSM only, followed by 500 epochs with the chosen loss combination. We use a batch size of 2048, Adam optimizer with learning rate 10^{-3}. We average results over two random training seeds. Metrics for sample quality and log-density estimation are computed on 4096 samples. The Fisher devergence and the classification objective are computed on the full time-grid. Sampling is performed using the DDIM denoising kernel Song et al. (2021a).

Stochastic Interpolant. For SIs, we use the linear interpolant $I_t(x_0, x_1) = (1-t)x_0 + tx_1$ and $\gamma : t \mapsto \sqrt{t(1-t)}$ bridging the 40-mode and 2-mode Gaussian mixtures described earlier. Time is discretized into 512 steps between 10^{-3} and $1 - 10^{-3}$. The potential is parameterized as

$$\mathrm{U}_{(\theta_1,\theta_2)}(t, x) = x^\top \mathrm{NN}^{\theta_1}(t, x) + \mathrm{NN}^{\theta_2}(t, x) \ ,$$

where $\mathrm{NN}^{\theta_1} : [0, T] \times \mathbb{R}^d \to \mathbb{R}^d$ and $\mathrm{NN}^{\theta_2} : [0, T] \times \mathbb{R}^d \to \mathbb{R}$ are MLPs with depth 4 and width 64 (if $d \leq 32$) or 256 otherwise. Time embeddings follow Song et al. (2021b). Training proceeds for 10k steps with DSM only, then 50k steps with the chosen objective. We use a batch size of 1024 and a learning rate of 5×10^{-4}, sampling endpoint distributions at each step. To reduce variance in $\mathcal{L}_{\text{DSM}}$ and $\mathcal{L}_{\text{C}t\text{SM}}$, we apply the antithetic trick (Albergo et al., 2023, Appendix 6.1). Results are averaged over two seeds, and evaluation metrics are computed on 4096 samples. The Fisher devergence and the classification objective are computed on the full time-grid.

H.2.3 ADDITIONAL RESULTS

In this section, we complete the results of Section 5 with additional metrics and problems.

Diffusion Model While Table 4 provides the same comparison as Table 3 but for the bi-modal case, Tables 5 and 7 examine how the number of classification levels affects log-density estimation, whereas Tables 6 and 8 focus on generation quality.

Stochastic Interpolant Like Figure 3, Figures 7a to 7d we visualize the determination coefficient R^2 between learned log-densities and the ground truth ones across time for each modal, as well as the log-density log-density scatter plots at two ends which should be a perfect diagonal line in optimality. In Tables 10 and 9, we report the quality of the estimated log-densities using the stochastic interpolant model.

H.3 COMPOSITION

In this section, we give the experimental details for the toy composition example from Section 5.

H.3.1 DISTRIBUTIONS DETAILS

Table 3: **Comparison on synthetic 40-mode Gaussian mixtures**. A DM with variance preserving noising scheme was trained on MOG-40 using the different objectives. We explore different values of the number of levels $N \in \{2, 4, 8, 16\}$ and ensure equal computational comparison between methods. We report the classification loss (10), Fisher divergence (FD), and Maximum Mean Discrepancy (MMD) ($\times 100$) from the denoising SDE. The classification approach matches DSM in Fisher divergence and MMD, while yielding markedly better consistency in classification loss.

	$\mathcal{L}_{\text{clf}} + \mathcal{L}_{\text{DSM}}$ (ours)			$\mathcal{L}_{C\text{tSM}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{\text{DSM}}$		
Dim	$\mathcal{L}_{\text{clf}}$	FD	MMD	$\mathcal{L}_{\text{clf}}$	FD	MMD	$\mathcal{L}_{\text{clf}}$	FD	MMD
8	$4.41_{\pm 0.12}$	$2.00_{\pm 1.48}$	$0.69_{\pm 0.59}$	$6.80_{\pm 0.86}$	$5.74_{\pm 2.21}$	$19.41_{\pm 0.77}$	$9.19_{\pm 0.33}$	$4.09_{\pm 3.89}$	$0.99_{\pm 0.64}$
16	$4.19_{\pm 0.14}$	$2.81_{\pm 1.38}$	$0.91_{\pm 0.32}$	$8.33_{\pm 2.36}$	$7.96_{\pm 2.11}$	$22.62_{\pm 0.45}$	$22.36_{\pm 0.76}$	$5.49_{\pm 5.23}$	$1.28_{\pm 0.56}$
32	$4.04_{\pm 0.23}$	$3.68_{\pm 1.47}$	$1.20_{\pm 0.44}$	$6.13_{\pm 1.45}$	$10.30_{\pm 1.95}$	$18.18_{\pm 1.51}$	$85.07_{\pm 9.53}$	$3.88_{\pm 3.49}$	$1.20_{\pm 0.42}$
64	$4.01_{\pm 0.46}$	$4.87_{\pm 1.95}$	$2.18_{\pm 1.02}$	$7.78_{\pm 1.64}$	$9.96_{\pm 1.84}$	$12.67_{\pm 3.66}$	$149.45_{\pm 33.76}$	$3.93_{\pm 3.48}$	$1.51_{\pm 0.15}$
128	$4.40_{\pm 1.00}$	$6.91_{\pm 2.47}$	$3.54_{\pm 1.34}$	$20.86_{\pm 4.93}$	$9.42_{\pm 1.82}$	$5.20_{\pm 0.34}$	$383.53_{\pm 35.99}$	$6.78_{\pm 5.94}$	$1.99_{\pm 0.35}$

Table 4: **Comparison of classification and score matching on synthetic Gaussian mixtures**. Mixtures with two modes are trained using the same architecture under DSM as well as conditional time-score matching and our classification objective, averaged over seeds and number of classification levels $N \in 2, 4, 8, 16$ (DSM uses the same number of score evaluations for every N). We report the classification loss (10), Fisher divergence, and MMD ($\times 100$) from the denoising SDE (all on 512 time-steps). The classification approach matches DSM in Fisher divergence and MMD, while yielding markedly better consistency in classification loss.

	$\mathcal{L}_{\text{clf}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{\text{CtSM}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{\text{DSM}}$		
Dim	$\mathcal{L}_{\text{clf}}$	FD	MMD	$\mathcal{L}_{\text{clf}}$	FD	MMD	$\mathcal{L}_{\text{clf}}$	FD	MMD
8	$4.14_{\pm 0.02}$	$2.48_{\pm 2.34}$	$6.94_{\pm 0.59}$	$5.55_{\pm 1.27}$	$6.78_{\pm 3.29}$	$20.45_{\pm 8.43}$	$17.88_{\pm 4.13}$	$1.21_{\pm 1.14}$	$5.91_{\pm 0.68}$
16	$3.95_{\pm 0.04}$	$3.47_{\pm 3.15}$	$8.57_{\pm 1.83}$	$17.97_{\pm 9.77}$	$9.15_{\pm 2.82}$	$22.50_{\pm 6.13}$	$191.78_{\pm 51.04}$	$0.83_{\pm 0.74}$	$7.13_{\pm 0.83}$
32	$3.84_{\pm 0.15}$	$4.86_{\pm 3.87}$	$11.91_{\pm 1.00}$	$27.05_{\pm 13.99}$	$10.54_{\pm 2.99}$	$28.59_{\pm 2.31}$	$194.54_{\pm 23.85}$	$1.04_{\pm 0.89}$	$8.62_{\pm 1.15}$
64	$3.83_{\pm 0.52}$	$4.39_{\pm 1.77}$	$15.30_{\pm 2.08}$	$47.65_{\pm 22.47}$	$11.57_{\pm 3.48}$	$27.49_{\pm 1.93}$	$208.32_{\pm 14.77}$	$1.16_{\pm 0.89}$	$10.73_{\pm 1.04}$
128	$3.85_{\pm 0.51}$	$6.86_{\pm 2.26}$	$17.61_{\pm 2.09}$	$151.48_{\pm 51.21}$	$17.25_{\pm 9.12}$	$30.69_{\pm 3.09}$	$1521.54_{\pm 538.48}$	$3.01_{\pm 2.30}$	$15.48_{\pm 0.57}$

Gaussian mixtures The Gaussian mixtures p_A and p_B in the left part of Figure 5 (top row) are both bi-modal. For p_A, the modes are centered at $\mu_1 = (-a, +a)$ and $\mu_2 = (+a, +a)$ with weights 0.3 and 0.7, respectively. For p_B, the modes are at $\mu_1 = (-a, -a)$ and $\mu_2 = (+a, -a)$ with weights 0.7 and 0.3. We chose $a = 0.5$. Both mixtures share identical covariances $\Sigma_1 = \Sigma_2 = 0.01 \times \mathrm{I}_2$.

Rings mixture The ring distribution is constructed as the product of a uniform distribution on $[0, 2\pi]$ and a Gaussian distribution on the radius with mean r and variance $\sigma^2 = 10^{-2}$ (with $r \gg \sigma$). Applying the polar transformation maps this distribution on $[0, 2\pi] \times \mathbb{R}^+$ into a ring shape in $\mathbb{R}^2$. In the left part of Figure 5 (bottom row), p_A is defined as a mixture of four such rings with radii $r \in \{1, 3, 5\}$, where each ring is assigned a weight proportional to its radius. This weighting makes the rings appear visually balanced in the mixture.

Uniforms mixture The distribution used as p_B in the bottom row of Figure 5 is an equilibrated mixture of 4 uniform distributions : $\mathcal{U}([-6.0, -1.6] \times [-1.4, 1.4])$, $\mathcal{U}([-1.4, 1.4] \times [1.6, 6.0])$, $\mathcal{U}([1.6, 6.0] \times [-1.4, 1.4])$ and $\mathcal{U}([-1.4, 1.4] \times [-6.0, -1.6])$.

H.3.2 Models training

The models follow the energy parameterization of Thornton et al. (2025), implemented with a 3-layer MLP of width 128 and trained under the variance-preserving noising scheme. The DSM baseline was trained for 500 epochs with a batch size of 4096, while `DiffCLF` was trained for the same number of epochs using the $N = 4$ version of the objective (10) with a batch size of 1024. Both models used a learning rate of 10^{-4} and were trained on a dataset of 100k samples. Figures 8 and 9 show samples generated via their respective denoising SDEs, demonstrating that both approaches successfully capture all target distributions.

Table 5: **Log-density estimation on synthetic 40-mode Gaussian mixtures.** We report classification loss, Fisher divergence, and average Effective Sample Size (ESS). The ESS is computed between the learned and exact log-densities using exact samples, averaged across time levels. Unlike Table 3, this table shows results for varying numbers of classification levels N. For fairness, each setting of N uses the same total number of score evaluations as $\mathcal{L}_{\text{DSM}}$.

		$\mathcal{L}_{\text{clf}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{\text{C}t\text{SM}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{\text{DSM}}$		
Dim	N	$\mathcal{L}_{\text{clf}}$	ESS	FD	$\mathcal{L}_{\text{clf}}$	ESS	FD	$\mathcal{L}_{\text{clf}}$	ESS	FD
8	2	4.61±0.04	89.9%±0.9%	3.61±0.25	5.14±0.08	88.2%±0.4%	5.27±0.08	8.80±0.13	90.1%±1.2%	5.45±0.83
8	4	4.40±0.03	93.2%±0.6%	2.50±0.00	5.48±0.19	90.6%±0.4%	4.69±0.01	9.06±0.05	90.4%±0.1%	4.18±0.20
8	8	4.32±0.00	94.2%±0.3%	1.20±0.03	6.77±0.48	91.9%±0.4%	4.09±0.00	9.20±0.05	91.6%±1.1%	3.94±0.11
8	16	4.31±0.00	96.2%±0.3%	0.69±0.01	7.83±0.16	92.8%±0.2%	3.92±0.04	9.68±0.08	91.5%±0.1%	2.78±0.08
16	2	4.40±0.05	85.8%±0.7%	4.15±0.11	5.17±0.42	77.7%±0.4%	8.77±0.26	22.36±0.55	84.7%±1.1%	6.83±0.71
16	4	4.22±0.04	86.8%±1.6%	3.37±0.04	11.40±2.85	78.4%±0.4%	8.33±0.17	21.45±0.36	85.8%±0.3%	7.13±1.34
16	8	4.09±0.00	88.8%±0.1%	2.26±0.12	10.04±1.60	80.8%±0.6%	7.52±0.29	22.93±0.46	87.7%±0.9%	4.64±0.89
16	16	4.05±0.00	91.5%±0.4%	1.48±0.01	5.48±0.18	81.6%±0.7%	7.11±0.03	22.69±0.61	88.9%±0.0%	3.38±0.14
32	2	4.41±0.04	76.8%±1.1%	4.70±0.11	5.66±0.11	67.8%±0.1%	10.19±0.10	73.34±0.94	76.4%±0.4%	4.88±0.38
32	4	4.03±0.07	77.1%±0.4%	4.04±0.03	6.63±0.38	70.7%±0.1%	9.46±0.35	81.87±3.99	81.9%±0.3%	4.43±0.10
32	8	3.89±0.00	79.7%±0.4%	3.33±0.06	6.46±0.26	71.4%±0.4%	9.64±0.43	86.24±2.31	80.4%±1.3%	3.49±0.24
32	16	3.83±0.01	84.7%±0.9%	2.65±0.03	13.92±2.81	72.7%±0.4%	9.32±0.06	98.82±1.74	82.9%±0.4%	2.72±0.01
64	2	4.68±0.34	66.0%±2.9%	5.88±0.13	21.25±2.64	55.9%±1.2%	10.10±0.23	121.14±12.99	69.5%±2.2%	4.86±0.68
64	4	4.04±0.10	69.0%±0.5%	5.16±0.03	62.11±47.56	55.8%±0.1%	10.42±0.02	121.76±3.21	71.3%±1.1%	4.00±0.09
64	8	3.70±0.04	72.8%±0.1%	4.42±0.10	34.83±5.43	56.5%±0.1%	10.17±0.03	152.74±1.00	72.6%±0.6%	3.57±0.03
64	16	3.61±0.01	72.3%±0.4%	4.05±0.03	85.59±68.72	57.8%±1.4%	10.47±0.27	202.18±4.46	75.1%±1.7%	3.28±0.18
128	2	5.98±0.05	54.1%±0.1%	7.30±0.06	11.80±2.10	44.7%±0.0%	9.33±0.00	427.77±12.04	60.0%±0.3%	8.34±0.43
128	4	4.52±0.11	54.6%±0.4%	7.39±0.04	17.12±1.23	44.8%±0.0%	9.33±0.01	383.60±15.45	62.6%±0.2%	6.88±0.13
128	8	3.63±0.07	55.8%±0.3%	6.79±0.01	54.49±28.78	45.0%±0.1%	9.40±0.01	356.38±39.38	61.9%±0.7%	5.83±0.33
128	16	3.47±0.01	55.0%±0.1%	6.17±0.01	24.84±9.45	45.1%±0.1%	9.51±0.10	366.36±16.04	62.6%±0.4%	6.09±0.05

Table 6: **Generation quality on 40-mode Gaussian mixtures.** We report Maximum Mean Discrepancy (MMD) Gretton et al. (2012) (×100), sliced 2-Wasserstein distance (×100), and total variation (TV) distance between mode-weight histograms (as in Noble et al. (2025)). Results are reported for varying classification levels N.

		$\mathcal{L}_{\text{clf}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{\text{C}t\text{SM}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{\text{DSM}}$		
Dim	N	MMD	Sliced W_2	TV	MMD	Sliced W_2	TV	MMD	Sliced W_2	TV
8	2	1.34±0.37	6.32±0.28	0.09±0.00	1.65±0.28	7.55±0.61	0.15±0.01	1.75±0.07	7.74±0.42	0.13±0.00
8	4	0.56±0.56	5.88±0.36	0.12±0.00	1.41±1.09	7.12±0.94	0.12±0.02	0.54±0.54	5.70±0.81	0.11±0.01
8	8	0.87±0.05	5.28±0.19	0.09±0.02	1.07±0.03	6.14±0.03	0.12±0.01	0.46±0.46	5.49±0.37	0.10±0.01
8	16	0.00±0.00	4.77±0.68	0.09±0.01	1.49±0.13	6.82±0.54	0.11±0.01	1.21±0.06	6.34±0.02	0.11±0.01
16	2	0.81±0.31	6.15±0.03	0.11±0.00	2.40±0.21	10.74±0.55	0.22±0.01	1.81±0.18	7.50±0.35	0.13±0.00
16	4	1.15±0.03	5.86±0.19	0.10±0.01	3.18±0.31	11.78±0.81	0.23±0.02	1.33±0.58	6.86±0.44	0.12±0.00
16	8	1.00±0.17	5.96±0.41	0.10±0.00	1.82±0.47	8.85±1.12	0.21±0.01	0.80±0.39	5.90±0.68	0.12±0.01
16	16	0.68±0.41	5.93±0.27	0.09±0.01	2.40±0.36	9.39±0.48	0.19±0.02	1.19±0.49	6.30±0.37	0.11±0.01
32	2	1.67±0.50	7.51±0.53	0.12±0.01	3.11±0.37	12.96±0.85	0.21±0.05	1.66±0.28	7.47±0.50	0.14±0.01
32	4	1.21±0.12	6.69±0.29	0.11±0.01	2.32±0.02	10.92±0.18	0.17±0.03	1.40±0.04	6.91±0.09	0.12±0.01
32	8	0.94±0.22	6.41±0.06	0.10±0.01	2.40±0.04	10.90±0.42	0.18±0.01	1.09±0.22	6.22±0.82	0.09±0.00
32	16	0.97±0.38	6.22±0.61	0.09±0.01	1.86±0.20	9.76±0.36	0.15±0.01	0.67±0.16	5.84±0.09	0.10±0.01
64	2	3.68±0.80	10.41±1.11	0.15±0.02	12.25±5.58	30.15±10.58	0.52±0.19	1.52±0.24	7.40±0.09	0.14±0.01
64	4	2.13±0.18	7.88±0.39	0.13±0.01	10.80±0.89	26.86±1.25	0.48±0.00	1.65±0.02	8.11±0.45	0.16±0.01
64	8	1.69±0.11	7.66±0.27	0.12±0.01	9.12±0.93	21.78±1.74	0.43±0.04	1.41±0.02	7.18±0.09	0.12±0.00
64	16	1.22±0.25	6.67±0.38	0.10±0.00	7.60±2.05	18.46±2.06	0.34±0.09	1.46±0.08	6.78±0.59	0.13±0.01
128	2	5.34±0.27	15.18±1.27	0.16±0.02	5.23±0.47	22.31±1.14	0.61±0.01	2.37±0.19	9.41±0.26	0.18±0.01
128	4	4.27±0.15	13.29±0.34	0.15±0.01	14.73±0.26	32.63±0.32	0.80±0.05	2.02±0.13	8.20±0.60	0.15±0.00
128	8	2.42±0.20	10.51±0.07	0.12±0.01	22.79±1.63	47.99±4.90	1.11±0.03	1.95±0.37	7.61±0.72	0.14±0.00
128	16	2.12±0.25	9.72±0.55	0.13±0.01	22.97±6.80	42.97±11.55	1.02±0.30	1.61±0.01	7.25±0.24	0.12±0.02

H.3.3 Composition algorithm details

A variety of training-free strategies have been proposed to compose diffusion models. Here, we focus on the "AND" operator, defined as the product distribution $p_{A,B} = p_A \times p_B$. The "OR" operator covered in Section 5 is given by $p_{A,B} = (1/2)p_A + (1/2)p_B$. As emphasized in Du et al.

Table 7: **Log-density estimation on synthetic 2-mode Gaussian mixtures.** We report classification loss, Fisher divergence, and average Effective Sample Size (ESS). The ESS is computed between the learned and exact log-densities using exact samples, averaged across time levels. Unlike Table 3, this table shows results for varying numbers of classification levels N. For fairness, each setting of N uses the same total number of score evaluations as $\mathcal{L}_{\text{DSM}}$.

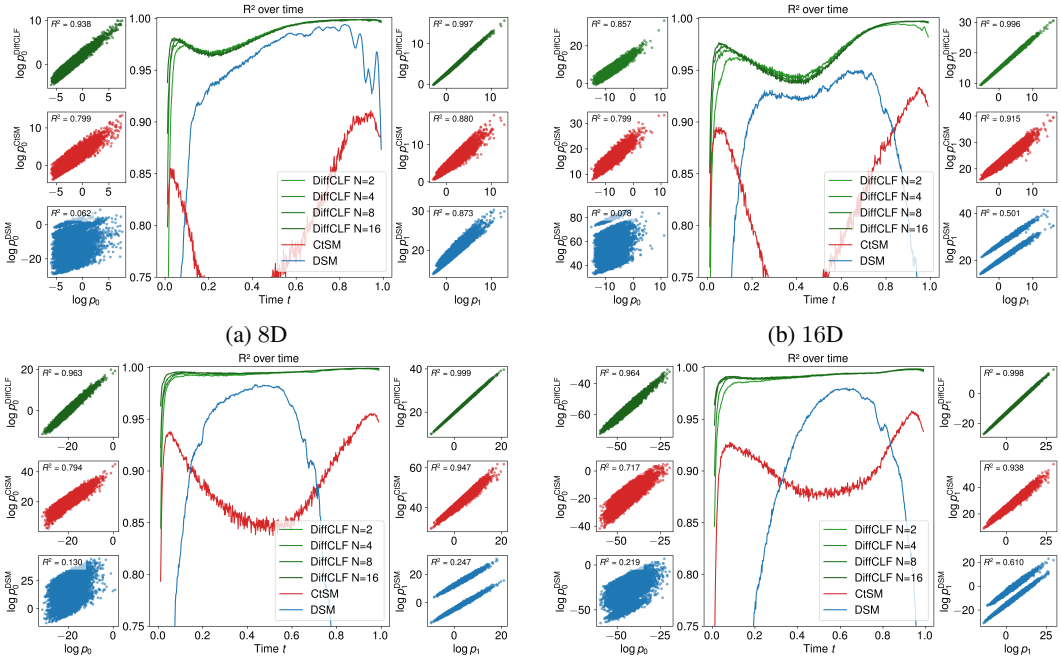
		$\mathcal{L}_{\text{clf}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{C\text{tSM}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{\text{DSM}}$		
Dim	N	$\mathcal{L}_{\text{clf}}$	ESS	FD	$\mathcal{L}_{\text{clf}}$	ESS	FD	$\mathcal{L}_{\text{clf}}$	ESS	FD
8	2	4.18±0.01	95.7%±2.3%	2.83±0.19	5.05±0.67	91.8%±0.1%	5.02±0.38	14.70±1.88	91.6%±0.9%	2.20±0.73
8	4	4.13±0.00	96.5%±0.4%	3.70±0.34	4.72±0.49	83.3%±1.5%	7.25±0.15	14.58±1.91	90.9%±1.5%	1.29±0.34
8	8	4.12±0.00	98.2%±0.0%	2.22±0.18	6.02±0.43	88.0%±8.3%	5.36±1.63	18.21±1.49	92.6%±1.2%	0.91±0.30
8	16	4.12±0.00	97.5%±0.1%	1.17±0.17	6.39±1.93	85.9%±1.8%	9.50±2.75	24.02±0.36	92.6%±1.0%	0.44±0.00
16	2	4.00±0.01	90.8%±4.0%	2.93±0.61	14.16±7.75	66.5%±2.3%	9.32±0.11	157.79±0.87	81.3%±0.3%	1.19±0.21
16	4	3.94±0.03	95.3%±0.5%	4.49±1.22	12.38±3.86	76.9%±3.6%	8.53±0.46	278.78±1.52	82.2%±1.5%	1.08±0.03
16	8	3.94±0.03	92.7%±1.3%	3.99±0.75	27.12±5.10	73.3%±9.8%	8.74±1.83	156.97±1.03	82.5%±0.5%	0.68±0.08
16	16	3.91±0.04	97.9%±0.4%	2.44±0.13	18.21±12.30	59.5%±1.5%	10.02±0.01	173.58±12.17	83.7%±0.3%	0.37±0.06
32	2	3.99±0.17	83.8%±3.9%	2.77±0.45	45.50±9.22	55.3%±6.1%	10.49±0.74	164.35±12.67	74.4%±0.9%	1.81±0.55
32	4	3.87±0.06	87.2%±6.5%	4.26±1.03	27.16±10.23	48.8%±0.5%	10.84±0.00	220.25±11.54	75.2%±0.2%	0.98±0.14
32	8	3.70±0.05	90.0%±1.2%	5.10±0.59	16.21±3.15	53.1%±4.5%	10.75±0.63	185.16±8.36	76.3%±0.1%	0.81±0.05
32	16	3.81±0.08	87.3%±0.8%	7.30±0.60	19.32±8.11	50.5%±2.0%	10.06±0.60	208.39±7.67	76.9%±0.8%	0.54±0.10
64	2	3.72±0.12	78.8%±1.2%	4.53±0.54	29.66±11.12	43.7%±0.4%	11.07±0.04	214.31±8.38	67.1%±0.5%	1.96±0.31
64	4	4.29±0.84	81.3%±0.0%	3.83±0.66	62.44±6.10	45.0%±0.9%	12.10±0.42	210.10±3.03	68.2%±0.4%	1.14±0.05
64	8	3.73±0.21	80.9%±6.9%	5.65±2.41	43.24±2.78	43.7%±1.7%	11.22±0.33	206.46±25.51	69.2%±0.4%	0.84±0.08
64	16	3.57±0.15	75.6%±7.0%	3.54±0.76	55.28±35.08	44.2%±0.2%	11.87±0.93	202.40±8.05	70.3%±0.6%	0.69±0.02
128	2	3.79±0.13	56.5%±0.7%	6.99±1.01	93.67±7.28	37.5%±0.1%	11.27±0.25	2192.68±397.10	56.3%±0.3%	4.27±0.34
128	4	4.66±0.14	61.5%±0.7%	5.89±0.50	141.52±40.48	36.8%±0.1%	28.36±13.03	1637.77±155.21	56.5%±0.3%	3.15±0.71
128	8	3.66±0.14	57.7%±1.6%	8.87±0.54	195.62±16.22	39.6%±2.9%	14.67±1.39	1401.30±161.64	56.8%±0.1%	1.95±0.06
128	16	3.30±0.03	62.4%±0.6%	5.71±0.41	175.09±50.85	37.0%±0.7%	14.69±1.52	854.42±168.79	56.8%±0.3%	2.66±0.30

Table 8: **Generation quality on 2-mode Gaussian mixtures.** We report Maximum Mean Discrepancy (MMD) Gretton et al. (2012) (×100), sliced 2-Wasserstein distance (×100), and total variation (TV) distance between mode-weight histograms (as in Noble et al. (2025)). Results are reported for varying classification levels N.

		$\mathcal{L}_{\text{clf}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{C\text{tSM}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{\text{DSM}}$		
Dim	N	MMD	Sliced W_2	TV	MMD	Sliced W_2	TV	MMD	Sliced W_2	TV
8	2	7.42±0.31	20.82±1.74	0.01±0.01	13.79±0.24	15.87±1.78	0.03±0.01	6.32±0.78	21.37±1.77	0.01±0.00
8	4	7.05±0.29	14.28±5.52	0.03±0.00	21.67±5.75	47.94±22.18	0.11±0.10	6.28±0.42	19.70±3.09	0.02±0.01
8	8	6.31±0.60	19.07±2.13	0.01±0.00	22.03±11.45	48.37±30.22	0.19±0.18	5.84±0.49	25.29±6.98	0.02±0.02
8	16	6.96±0.47	13.43±2.67	0.02±0.01	24.33±7.51	59.01±25.17	0.19±0.13	5.18±0.01	13.89±0.77	0.01±0.01
16	2	9.93±2.64	15.48±1.49	0.02±0.00	21.18±7.41	68.65±16.57	0.18±0.06	7.63±0.78	11.31±7.99	0.00±0.00
16	4	8.56±1.09	22.40±0.20	0.02±0.01	22.40±5.68	46.29±5.80	0.08±0.03	7.69±0.43	22.83±6.96	0.04±0.01
16	8	8.56±1.22	10.92±0.28	0.01±0.01	19.38±1.92	48.88±25.33	0.06±0.00	6.70±0.89	22.80±9.18	0.04±0.02
16	16	7.25±0.26	13.33±6.27	0.01±0.00	27.06±5.19	115.65±38.93	0.31±0.15	6.50±0.09	17.29±4.43	0.03±0.01
32	2	11.98±0.45	18.86±4.51	0.02±0.01	26.50±1.32	76.92±6.78	0.37±0.01	9.99±0.06	28.01±0.00	0.03±0.01
32	4	12.27±1.72	18.98±5.11	0.03±0.00	30.14±0.64	85.84±0.70	0.57±0.03	9.50±0.17	14.84±2.00	0.01±0.00
32	8	11.19±0.07	11.49±3.99	0.03±0.01	28.07±3.31	76.61±3.20	0.54±0.09	7.29±0.05	10.69±3.76	0.00±0.00
32	16	12.22±0.30	9.63±3.58	0.01±0.00	29.68±0.20	85.34±2.66	0.63±0.03	7.71±0.15	26.74±4.60	0.05±0.02
64	2	15.68±1.39	20.10±3.14	0.01±0.00	27.41±2.15	87.37±3.11	0.64±0.01	11.62±1.10	19.19±9.18	0.03±0.00
64	4	15.06±1.70	22.11±12.00	0.03±0.03	27.12±1.68	86.51±1.15	0.51±0.03	11.21±0.29	18.76±5.50	0.00±0.00
64	8	15.39±2.67	21.27±3.64	0.02±0.01	27.09±2.34	91.65±4.87	0.67±0.00	10.44±0.86	18.83±0.14	0.02±0.01
64	16	15.07±2.26	23.33±8.53	0.03±0.02	28.35±0.95	83.80±2.62	0.55±0.12	9.63±0.06	19.02±8.67	0.02±0.01
128	2	17.21±0.47	31.28±8.20	0.01±0.00	29.11±0.04	82.95±0.86	0.67±0.00	15.72±0.52	34.61±6.61	0.04±0.02
128	4	16.78±0.69	32.77±0.65	0.06±0.01	31.45±0.19	82.31±6.71	0.66±0.01	15.84±0.57	15.26±2.56	0.02±0.01
128	8	19.74±3.16	49.72±10.95	0.07±0.03	32.18±5.65	92.22±1.62	0.66±0.00	14.99±0.25	31.94±13.94	0.01±0.01
128	16	16.69±0.72	39.50±0.44	0.05±0.01	30.01±0.74	88.74±7.63	0.64±0.02	15.38±0.42	22.36±7.18	0.02±0.02

(2023), if $p_A(t,\cdot)$ and $p_B(t,\cdot)$ denote the time-dependent marginals obtained by noising p_A and p_B, then their product $p_A(t,\cdot) \times p_B(t,\cdot)$ is not equal to the marginal of the noised operator $p_{A,B}$. Thus, we do not obtain the correct sequence of marginals "for free". Nevertheless, this construction defines a valid interpolation: it recovers $p_{A,B}$ at $t = 0$ and approaches a Gaussian at $t = T$.

Based on this observation, Du et al. (2023) proposed annealed Langevin sampling over this sequence, i.e., running an MCMC chain at each noise level, with improved performance when de-



(a) 8D (b) 16D

(c) 32D (d) 64D

Figure 7: R^2 **of learned versus exact log-densities for SIs between MoG-40 and MoG-2 across different dimensions.** Complementing Figure 3, which shows detailed 2D scatter plots, this figure demonstrates that `DiffCLF` maintains consistently higher R^2 as dimensionality increases.

Table 9: **Log-density focused metrics for stochastic interpolants.** We report classification loss, time-average ESS, and Fisher divergence for a SI model learned between MOG-40 and MOG-2. The results present different values of N but the computational budgets remain equal across methods.

		$\mathcal{L}_{\text{clf}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{\text{CtSM}} + \mathcal{L}_{\text{DSM}}$			$\mathcal{L}_{\text{DSM}}$		
Dim	N	$\mathcal{L}_{\text{clf}}$	ESS	FD	$\mathcal{L}_{\text{clf}}$	ESS	FD	$\mathcal{L}_{\text{clf}}$	ESS	FD
8	2	5.26±0.00	91.45%±0.02%	0.16±0.00	-	-	-	-	-	-
8	4	5.24±0.00	91.78%±0.08%	0.16±0.00	-	-	-	-	-	-
8	8	5.23±0.00	91.83%±0.07%	0.16±0.00	-	-	-	-	-	-
8	16	5.23±0.00	91.96%±0.20%	0.16±0.00	6.54±0.01	5.10%±0.11%	0.96±0.00	6.85±0.01	76.91%±0.07%	0.39±0.00
16	2	4.87±0.00	68.08%±0.14%	0.22±0.00	-	-	-	-	-	-
16	4	4.86±0.00	70.26%±0.10%	0.23±0.00	-	-	-	-	-	-
16	8	4.85±0.00	69.73%±0.14%	0.23±0.00	-	-	-	-	-	-
16	16	4.85±0.00	69.44%±0.13%	0.23±0.00	8.97±0.01	2.74%±0.01%	1.04±0.00	7.33±0.00	46.93%±0.01%	0.46±0.00
32	2	4.47±0.00	88.76%±0.02%	0.06±0.00	-	-	-	-	-	-
32	4	4.46±0.00	90.52%±0.12%	0.05±0.00	-	-	-	-	-	-
32	8	4.46±0.00	91.36%±0.08%	0.05±0.00	-	-	-	-	-	-
32	16	4.46±0.00	91.87%±0.01%	0.05±0.00	11.62±0.02	1.35%±0.16%	0.61±0.00	5.75±0.00	48.72%±0.10%	0.30±0.00
64	2	4.12±0.00	71.00%±0.04%	0.08±0.00	-	-	-	-	-	-
64	4	4.12±0.00	75.26%±0.26%	0.07±0.00	-	-	-	-	-	-
64	8	4.12±0.00	74.01%±0.06%	0.07±0.00	-	-	-	-	-	-
64	16	4.12±0.00	74.42%±0.34%	0.07±0.00	16.52±0.05	0.76%±0.03%	0.43±0.00	5.67±0.02	28.38%±0.14%	0.31±0.00
128	2	3.81±0.00	54.51%±0.09%	0.08±0.00	-	-	-	-	-	-
128	4	3.79±0.00	47.37%±0.02%	0.07±0.00	-	-	-	-	-	-
128	8	3.78±0.00	50.34%±0.07%	0.07±0.00	-	-	-	-	-	-
128	16	3.78±0.00	52.65%±0.22%	0.06±0.00	21.33±0.00	0.23%±0.00%	0.38±0.00	67.03±0.12	12.35%±0.07%	0.32±0.00

noising steps (via the discretized SDE equation 8 using $\nabla \log p_A(t, \cdot) + \nabla \log p_B(t, \cdot)$) are inserted between MCMC transitions. Building on this, Thornton et al. (2025) suggested using the discretized denoising dynamics, (lthough formally incorrect, as noted above) as a proposal distribution within a Sequential Monte Carlo (SMC) (Doucet et al., 2001; Del Moral et al., 2006) framework.

Table 10: **Averaged log-density metrics for stochastic interpolants.** We report classification loss and Fisher divergence across different dimensions. The different computation budgets of Table 9 are here averaged.

	$\mathcal{L}_{\text{clf}} + \mathcal{L}_{\text{DSM}}$		$\mathcal{L}_{\text{C}t\text{SM}} + \mathcal{L}_{\text{DSM}}$		$\mathcal{L}_{\text{DSM}}$	
Dim	Classif	Fisher	Classif	Fisher	Classif	Fisher
8	$5.35_{\pm 0.07}$	$4.38_{\pm 0.77}$	$8.59_{\pm 0.16}$	$40.69_{\pm 1.88}$	$8.03_{\pm 0.44}$	$0.37_{\pm 0.06}$
16	$4.88_{\pm 0.02}$	$5.21_{\pm 0.93}$	$16.10_{\pm 0.46}$	$42.04_{\pm 2.17}$	$10.15_{\pm 0.82}$	$0.52_{\pm 0.09}$
32	$4.67_{\pm 0.10}$	$5.63_{\pm 0.97}$	$30.88_{\pm 2.49}$	$40.60_{\pm 2.58}$	$17.69_{\pm 2.56}$	$0.53_{\pm 0.08}$
64	$4.17_{\pm 0.02}$	$2.71_{\pm 0.51}$	$56.54_{\pm 2.91}$	$35.63_{\pm 2.57}$	$9.36_{\pm 2.82}$	$0.23_{\pm 0.04}$
128	$3.84_{\pm 0.03}$	$2.66_{\pm 0.44}$	$128.28_{\pm 8.23}$	$279.37_{\pm 3.91}$	$4.54_{\pm 0.28}$	$0.24_{\pm 0.04}$

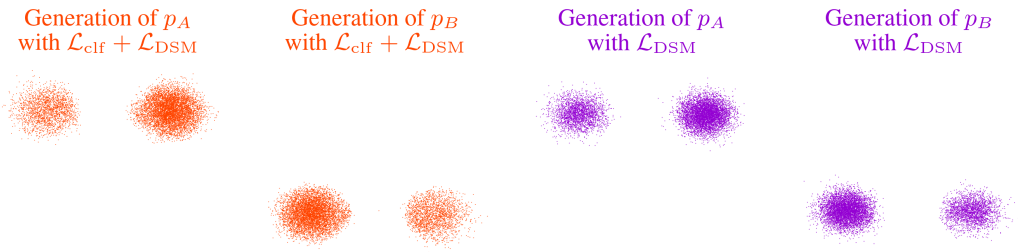


Figure 8: **Samples generated by the models trained on the p_A and p_B distribution for the "OR" composition task.** 8192 samples are displayed obtained by discretization of the denoising SDE (8) using the exponential integrator for 512 steps.

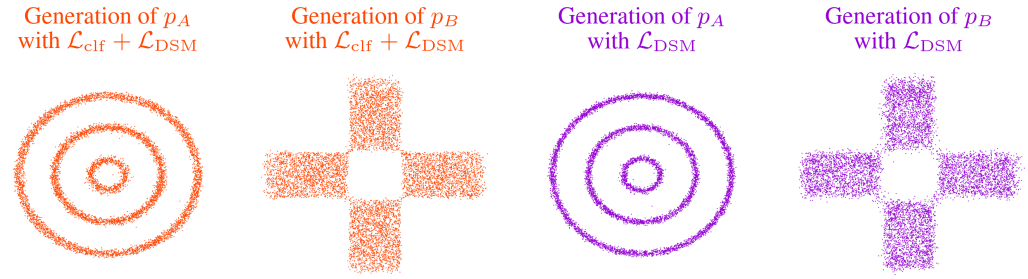


Figure 9: **Samples generated by the models trained on the p_A and p_B distribution for the "AND" composition task.** 8192 samples are displayed obtained by discretization of the denoising SDE (8) using the exponential integrator for 512 steps.

In this work, we combine the strengths of both approaches: we apply standard SMC (Algorithm 1) to the sequence $p_k(\cdot) = p_A(t_k, \cdot) \times p_B(t_k, \cdot)$, using the Metropolis-adjusted Langevin algorithm (MALA) (Roberts & Tweedie, 1996) as the transition kernel. This strategy retains the theoretical guarantees of SMC without depending on the incorrect denoising kernel, which we found prone to divergence. Concretely, we run $N = 64$ MALA steps at each level, tuning the step size to maintain a 75% acceptance rate, and perform adaptive resampling with a threshold of $\alpha = 30\%$ (Chopin & Papaspiliopoulos, 2020).

H.4 Boltzmann Generators

In this section, we describe how diffusion models or stochastic interpolants can be integrated into annealed sampling methods to build unbiased estimators. Let ρ denote a simple base distribution (e.g., a Gaussian), and π the target distribution, known up to a normalizing constant. Our goal is to compute expectations $\mathbb{E}_\pi[\phi(Y)]$ for a π-measurable test function ϕ.

Algorithm 1 Sequential Monte Carlo (SMC) algorithm with adaptive resampling

Input: Sequence of target distributions $\{p_k\}_{k=0}^{K}$, number of MCMC steps per level $L \geq 1$, number of particles $N \geq 1$, ESS resampling threshold $\alpha \in (0, 1]$

▷ Initialization

$Y_K^{1:N} \overset{\text{iid}}{\sim} p_K$, $\tilde{W}_K^{1:N} = 1/N$

for $k \leftarrow K-1$ **to** 0 **do**

  ▷ 1. Compute the SMC importance weights (in parallel, for $n = 1, \ldots, N$)

  $\tilde{W}_k^n \leftarrow (p_k(Y_{k+1}^n)/p_{k+1}(Y_{k+1}^n))\, \tilde{W}_{k+1}^n$

  ▷ 2. Compute the accumulated ESS (in parallel, for $n = 1, \ldots, N$)

  $\mathrm{ESS}_k \leftarrow \left(\sum_{n=1}^N \tilde{W}_k^n\right)^2 / \left(\sum_{n=1}^N (\tilde{W}_k^n)^2\right)$

  ▷ 3. Resample $X_k^{1:N}$ if the ESS too low

  if $\mathrm{ESS}_{k+1} < \alpha N$ **then**

    ▷ Normalize the importance weights (in parallel, for $n = 1, \ldots, N$)

    $W_k^n \leftarrow \tilde{W}_k^n / \sum_{j=1}^N \tilde{W}_k^j$

    ▷ Resample the particles

    $I^{1:N} \sim \mathcal{M}(W_k^1, \ldots, W_k^N)$

    $X_k^{1:N} \leftarrow Y_{k+1}^{I^{1:N}}$, $\tilde{W}_{k+1}^{1:N} = 1/N$

  end

  else

    $X_k^n \leftarrow Y_{k+1}^n$

  end

  ▷ 4. Sample from p_k by starting from sample Y_k^n (in parallel, for $n = 1, \ldots, N$)

  $\tilde{Y}_k^{1:L} \leftarrow \mathrm{MCMC}(p_k, L, \delta_{X_k^n})$

  $Y_k^n \leftarrow \tilde{Y}_k^L$

end

Output: Sequence of samples $Y_{0:K}^{1:N}$ such that $Y_k^{1:N} \overset{\text{iid}}{\sim} p_k$ (approximately)

Importance Sampling (IS). A natural approach is *Importance Sampling (IS)*. Provided $\mathrm{supp}(\rho) \subseteq \mathrm{supp}(\pi)$,

$$\mathbb{E}_\pi[\phi(Y)] = \int \phi(y)\pi(y)\mathrm{d}y = \mathbb{E}_\rho[w(Y)\phi(Y)] \quad w = \pi/\rho\,. \tag{114}$$

This yields the Monte Carlo estimator

$$\mathbb{E}_\pi[\phi(Y)] \approx \frac{1}{N}\sum_{i=1}^N W_i\phi(Y_i), \quad Y_i \sim \rho\,,$$

where $\{Y_i\}_{i=1}^N$ are called particles. When π is unnormalized, we use normalized weights $\tilde{W}_i = W_i/\sum_j W_j$, leading to a biased but consistent estimator. The main drawback of IS is variance explosion when ρ poorly overlaps with π, particularly in high dimensions (Agapiou et al., 2017).

Annealed Importance Sampling (AIS). To alleviate mismatch between ρ and π, Neal (2001) introduced *Annealed Importance Sampling* (AIS). AIS extends the problem to a sequence of distributions and defines an augmented target–proposal pair

$$\bar{\pi}(y_{0:K}) = \pi(x_0)\prod_{k=0}^{K-1} q_{k+1|k}(y_{k+1}|y_k), \quad \bar{\rho}(y_{0:K}) = \rho(x_K)\prod_{k=0}^{K-1} q_{k|k+1}(y_k|y_{k+1})\,, \tag{115}$$

where $\{q_{k+1|k}\}_{k=0}^{K-1}$ and $\{q_{k|k+1}\}_{k=0}^{K-1}$ are forward and backward Markov kernels, often chosen as reversible MCMC kernels (see Neal (2001)). Expectations under π can then be expressed as expectations under $\bar{\pi}$ and estimated via IS with proposal $\bar{\rho}$

$$\mathbb{E}_\pi[\phi(Y)] = \int \phi(y_0)\bar{\pi}(y_{0:K})\mathrm{d}y_{0:K} \approx \frac{1}{N}\sum_{i=1}^N \bar{W}^i\phi(Y_0^i), \quad (Y_0^i, \ldots, Y_K^i) \sim \bar{\rho}\,. \tag{116}$$

where $\bar{W}^i = \bar{\pi}(Y_{0:K}^i)/\bar{\rho}(Y_{0:K}^i)$. AIS thus interpolates between ρ and π by gradually refining proposals through intermediate distributions.

Sequential Monte Carlo (SMC). A challenge in AIS is *weight degeneracy*: as k increases, particles sampled from $\bar{\rho}$ may diverge from the marginals of $\bar{\pi}$, especially in high dimensions. *Sequential Monte Carlo* (SMC) (Doucet et al., 2001; Del Moral et al., 2006) addresses this by introducing a sequence of intermediate distributions p_k with tractable unnormalized densities which aim to approximate the marginals of $\bar{\pi}$ (i.e., $p_k(y_k) \approx \int \bar{\pi}(y_{0:K})\mathrm{d}y_{0:k-1}\mathrm{d}y_{k:K}$). SMC proceeds by:

1. running AIS between ρ and p_k,
2. resampling particles to match p_k using the previous AIS approximation, and
3. running AIS between p_k and π.

This resampling step prevents degeneracy and improves stability. While early methods performed resampling at every step, modern implementations use adaptive criteria to trigger resampling only when needed (Chopin & Papaspiliopoulos, 2020). The algorithm is presented in Algorithm 1 in the classic case where the forward and backward kernels are the same p_k-reversible MCMC kernel (which simplifies the weights) and the generic one is in Algorithm 2.

SMC in DMs and SIs. When dealing with a stochastic process such as Equation (1), a natural construction for annealed methods is to take the conditional distributions of $Y_{t_{k+1}}|Y_{t_k}$ as forward kernels, $Y_{t_k}|Y_{t_{k+1}}$ as backward kernels, and the marginals of Y_{t_k} as intermediate distributions p_k, with $Y_{t_0} \sim \pi$ and $Y_{t_K} \sim \rho$. DMs and SIs provide this setup by design, as their dynamics satisfy the endpoint conditions. In the case of DMs, Zhang et al. (2025a) propose using the exact noising kernel as forward kernel and the discretized denoising SDE (8) as the backward kernel in AIS, requiring only the score. Extending this principle to SMC, Phillips et al. (2024) additionally incorporate approximations of the marginals, precisely the focus of this work. In our experiments (Section 5), we apply the same idea to SIs: the forward and backward kernels are discretizations of SDE (9), both depending only on the score, while the marginal approximations required by SMC are learned either via DSM (7) or `DiffCLF` (10) (with $N = 2$ levels). Precisely, in Algorithm 2, we set $p_k(\cdot) = p^\theta_{t_k}(\cdot)$ and, given v^θ an approximation of v, we denote $b^\theta_t(\cdot) = v^\theta_t(\cdot) - \left[\dot{\gamma}(t)\gamma(t) + g(t)^2/2)\right] \nabla \log p^\theta_t(\cdot)$ and set

$$p_{k+1|k}(\cdot|y_k) = \mathcal{N}\left(y_k + (t_{k+1} - t_k)b^\theta_{t_k}(y_k),\ g(t_k)^2(t_{k+1} - t_k)\mathrm{I}_d\right) ,$$
$$p_{k|k+1}(\cdot|y_{k+1}) = \mathcal{N}\left(y_{k+1} - (t_{k+1} - t_k)b^\theta_{t_{k+1}}(y_{k+1}),\ g(t_{k+1})^2(t_{k+1} - t_k)\mathrm{I}_d\right) ,$$

with $g(\cdot) = 10^{-2}$. We run SMC with $N = 8192$ particles and adaptive resampling.

H.4.1 Computation of metrics

The same neural networks used in the analytical comparison with DSM on the MOG benchmarks are employed here, with a single modification: whenever the model is evaluated at $t = 0$, its output is replaced by the exact target density. Regarding the metrics, we report the sliced Wasserstein distance and the sliced Kolmogorov-Smirnov which are defined between two distributions p and q as

$$\text{s-}W_2(p, q) = \sqrt{\int_{\mathbb{S}^{d-1}} \int_0^1 \left|\mathrm{F}^{-1}_{p_\theta}(s) - \mathrm{F}^{-1}_{q_\theta}(s)\right| \mathrm{d}s\mathrm{d}\theta} ,$$
$$\text{s-KS}(p, q) = \int_{\mathbb{S}^{d-1}} \sup_{s\in\mathbb{R}} \left|\mathrm{F}_{p_\theta}(s) - \mathrm{F}_{q_\theta}(s)\right| \mathrm{d}s\mathrm{d}\theta ,$$

where p_θ (respectively q_θ) is the push-forward of p (respectively q) through $x \mapsto x^T\theta$ and $\mathrm{F}_\cdot$ indicates the cumulative distribution function. Those distances can be seen as functions of expectations with respect to p and q as by definition, for any distribution μ, $\mathrm{F}_\mu(x) = \mathbb{E}_\mu\left[\mathbf{1}_{Y\leq x}\right]$.

In the BG experiments, we approximate the distances between a target distribution π and itself by setting $\mathrm{F}_p = \mathrm{F}_\pi$ and replacing F_q by a Monte-Carlo approximation of F_π.

H.5 Free energy estimation

In this section, we provide addtional background on free energy difference estimation and *Thermodynamics Integration* (TI) based on (Máté et al., 2025), and the experimental details for free energy estimation.

Algorithm 2 Extension of the Sequential Monte Carlo algorithm for DMs and SIs

Input: Sequence of target distributions $\{p_k\}_{k=0}^K$, forward kernels $\{q_{k+1|k}\}_{k=0}^{K-1}$, backward kernels $\{q_{k|k+1}\}_{k=0}^{K-1}$, number of MCMC steps per level $L \geq 1$, number of particles $N \geq 1$, ESS resampling threshold $\alpha \in (0, 1]$

▷ Initialization

$Y_K^{1:N} \overset{\text{iid}}{\sim} p_K$, $\tilde{W}_K^{1:N} = 1/N$

for $k \leftarrow K-1$ **to** 0 **do**

  ▷ 1. Move to the next distribution with backward kernel (in parallel, for $n = 1, \ldots, N$)

  $\tilde{X}_{k+1}^n \sim q_{k|k+1}(\cdot|Y_{k+1}^n)$

  ▷ 2. Compute the SMC importance weights (in parallel, for $n = 1, \ldots, N$)

  $\tilde{W}_k^n \leftarrow \frac{p_k(\tilde{X}_k^n) p_{k+1|k}(Y_{k+1}^n|\tilde{X}_k^n)}{p_{k+1}(Y_{k+1}^n) q_{k|k+1}(\tilde{X}_{k+1}^n|Y_{k+1}^n)} \tilde{W}_{k+1}^n$

  ▷ 3. Compute the accumulated ESS (in parallel, for $n = 1, \ldots, N$)

  $\text{ESS}_k \leftarrow \left(\sum_{n=1}^N \tilde{W}_k^n\right)^2 / \left(\sum_{n=1}^N (\tilde{W}_k^n)^2\right)$

  ▷ 4. Resample $Y_{k+1}^{1:N}$ if the ESS too low

  if $\text{ESS}_{k+1} < \alpha N$ **then**

    ▷ Normalize the importance weights (in parallel, for $n = 1, \ldots, N$)

    $W_k^n \leftarrow \tilde{W}_k^n / \sum_{j=1}^N \tilde{W}_k^j$

    ▷ Resample the particles

    $I^{1:N} \sim \mathcal{M}(W_k^1, \ldots, W_k^N)$

    $X_k^{1:N} \leftarrow Y_{k+1}^{I^{1:N}}$, $\tilde{W}_{k+1}^{1:N} = 1/N$

  end

  else

    $X_k^n \leftarrow Y_{k+1}^n$

  end

  ▷ 5. Sample from p_k by starting from sample Y_k^n (in parallel, for $n = 1, \ldots, N$)

  $\tilde{Y}_k^{1:L} \leftarrow \text{MCMC}(p_k, L, \delta_{X_k^n})$

  $Y_k^n \leftarrow \tilde{Y}_k^L$

end

Output: Sequence of samples $Y_{0:K}^{1:N}$ such that $X_k^{1:N} \overset{\text{iid}}{\sim} p_k$ (approximately)

H.5.1 Free energy difference estimation with Thermodynamics Integration

As illustrated in Appendix C.2, we aim to estimate the free energy difference between two states A and B, which are associated with energy/potential functions U_A, U_B respectively. The free energy difference between these two states is the difference between the log partition functions

$$\Delta \mathrm{F}_{AB} = \mathrm{F}_A - \mathrm{F}_B = \log \frac{Z_B}{Z_A}.$$

TI leverages a sequence of intermediate distributions $(p_t)_{t\in[0,1]}$ where $p_0 = p_A$ and $p_1 = p_B$ to estimate the free energy difference via

$$\begin{aligned}
\Delta F_{AB} &= \int_0^1 \partial_t \log Z_t dt \\
&= \int_0^1 \frac{\partial_t Z_t}{Z_t} dt \\
&= \int_0^1 \frac{\partial_t \int \exp(-U_t(x)) dx}{Z_t} dt \\
&= \int_0^1 \frac{-\int \exp(-U_t(x)) \partial_t U_t(x) dx}{Z_t} dt \\
&= -\int_0^1 \mathbb{E}_{p_t}\left[\partial_t U_t(x)\right] dt. \qquad (117)
\end{aligned}$$

Furthermore, Máté et al. (2025) proposes the define the intermediate distributions via SIs and using U_t^θ. In such case, the estimation becomes

$$\Delta F_{AB} = -\int_0^1 \mathbb{E}_{p_t^\theta}\left[\partial_t \mathrm{U}_t^\theta(x)\right] dt,$$

which has two issues:

(1) Difficult to draw samples from p_t^θ,

(2) the boundaries are not match, i.e. $\mathrm{U}_0^\theta \neq U_A$ and $\mathrm{U}_1^\theta \neq U_B$.

To alleviate the first issue, one could approximate $\mathbb{E}_{p_t^\theta}$ with $\mathbb{E}_{p_t}$. To fix the second issue, one could estimate the free energy difference between U_A and U_0^θ ($\Delta F_{U_A,\mathrm{U}_0^\theta}$) and U_B and U_1^θ ($\Delta F_{\mathrm{U}_1^\theta,U_B}$) via FEP (Equation (47)) described in Appendix C.2. Hence, the final estimation of ΔF_{AB} is given by:

$$\Delta F_{AB} = \Delta F_{U_A,\mathrm{U}_0^\theta} \underbrace{- \int_0^1 \mathbb{E}_{p_t^\theta}\left[\partial_t \mathrm{U}_t^\theta(x)\right] dt}_{\Delta F_{\mathrm{U}_0^\theta,\mathrm{U}_1^\theta}} + \Delta F_{\mathrm{U}_1^\theta,U_B}. \tag{118}$$

H.5.2 Experimental details

Alanine dipeptide details. Both ALDP-imp and ALDP-vac are defined with AMBER ff96 classical force field, under temperature $300K$. For ALDP-imp, we use the samples generated from Midgley et al. (2023), which contains $1,000,000$ samples and we subsample $250,000$ of them as dataset. To gather equilibrium samples from ALDP-vac, we simulate MD with `openmmtools` (Chodera et al., 2025) to generate $250,000$ samples, where we follow hyperparameter choices as (He et al., 2025a, Appendix D.1.3).

Stochastic Interpolant. For SIs, we bridge ALDP-imp (p_0) and ALDP-vac (p_1) with the linear interpolant $I_t(x_0, x_1) = (1-t)x_0 + tx_1$ and $\gamma : t \mapsto \sqrt{0.01t(1-t)}$. Time is discretized into 512 steps between 10^{-3} and $1 - 10^{-3}$. The potential is parameterized as

$$\mathrm{U}_\theta(t, x) = x^\top \mathrm{NN}_\theta(t, x) \ ,$$

where $\mathrm{NN}_\theta : [0, T] \times \mathbb{R}^d \to \mathbb{R}^d$ is an EGNN with depth 5, width 128, sinusoidal time embeddings, learnable bond embeddings, and learnable atom-type embeddings. Time embeddings follow Song et al. (2021b). Training proceeds for 500 epochs with DSM only, then 500 epochs with the chosen objective. We use a batch size of 512 and a learning rate of 5×10^{-4}, sampling endpoint distributions at each step. To reduce variance in $\mathcal{L}_{\mathrm{DSM}}$, we apply the antithetic trick Albergo et al. (2023).

Architecture, training and evaluation details. We use exactly the same network architecture as illustrated in He et al. (2025a). To enhance the energy learning, we follow Máté et al. (2025); He et al. (2025a) to apply *Target Score Matching* (TSM) (Bortoli et al., 2024),

$$\mathcal{L}_{\mathrm{TSM}}^{\mathrm{imp}}(\theta) = \mathbb{E}_{t\sim U(0,0.5)}\mathbb{E}_{y_0,y_1}\mathbb{E}_{y_t|y_0,y_1}\left[\left\|\nabla \log p_t^\theta(y_t) - U_{\mathrm{imp}}(y_0)\right\|^2\right], \tag{119}$$

$$\mathcal{L}_{\mathrm{TSM}}^{\mathrm{vac}}(\theta) = \mathbb{E}_{t\sim U(0.5,1)}\mathbb{E}_{y_0,y_1}\mathbb{E}_{y_t|y_0,y_1}\left[\left\|\nabla \log p_t^\theta(y_t) - U_{\mathrm{vac}}(y_1)\right\|^2\right], \tag{120}$$

where U_{imp} and U_{vac} are the energy function for the implicit-solvent and vacuum defined with AMBER ff96 classical force field respectively. Máté et al. (2025) uses

$$\mathcal{L}_{\mathrm{base}} := \mathcal{L}_{\mathrm{dsm}} + \mathcal{L}_{\mathrm{TSM}}^{\mathrm{imp}} + \mathcal{L}_{\mathrm{TSM}}^{\mathrm{vac}}$$

to train energy-based models for TI. To evaluate our method, we simply add the diffusive classification loss to the aforementioned base loss, i.e.

$$\mathcal{L}_{\mathrm{base}} + \mathcal{L}_{\mathrm{clf}},$$

where we use $N = 4$ classes in $\mathcal{L}_{\mathrm{clf}}$

Estimation details. We use Equation (118) to estimate the free energy difference. In particular, we draw 5000 samples from ALDP-imp and 5000 samples from ALDP-vac, and use FEP (Equation (47)) to estimate $\Delta F_{U_{\mathrm{imp}},\mathrm{U}_0^\theta}$ and $\Delta F_{\mathrm{U}_1^\theta,U_{\mathrm{vac}}}$. To estimate $\Delta F_{\mathrm{U}_0^\theta,\mathrm{U}_1^\theta}$, we uniformly discretize the time from 0 to 1 with 1000 steps and draw 5000 samples from each marginal distributions p_t.

H.6 ADDITIONAL EXPERIMENTS ON MOLECULAR SYSTEMS

In this section, we present additional experiments on training energy-based diffusion models $(p_t^\theta)_t$ for molecular systems, including the Müller-Brown potential, Alanine Dipeptide (ALDP), and Chignolin, by training only with samples from the equilibrium distribution and without any access to their energies or forces. We simply compare classic DSM training and our method, i.e. jointly training with DSM and the diffusive classification loss. To evaluate the trained model, we simulate molecular dynamics using only $p_{t=0}^\theta$ and compare the simulated samples.

Setup. We follow exactly the same settings of Plainer et al. (2025), for both dataset processing, network architecture, and hyperparameters. Details of the three different systems could be found in the Appendix B in Plainer et al. (2025). As a result of lack of permission to the ground-truth dataset of Chignolin, we use the samples generated from Plainer et al. (2025) as a data set, which is of size 50000, and train both models. For simulations based on learn models $p_{t=0}^\theta$, we follow Plainer et al. (2025) to use Langevin dynamics and use exactly the same hyperparameters.

Results. We train energy-based Diffusion models for the aforementioned three different molecular systems, with DSM only ($\mathcal{L}_{\text{dsm}}$) and `DiffCLF` ($\mathcal{L}_{\text{dsm}} + \mathcal{L}_{\text{clf}}$), where we use $N = 4$ for $\mathcal{L}_{\text{clf}}$. Figure 10 demonstrates the effectiveness of the proposed diffusive classification loss, which help learning better energy at $t = 0$ while not bringing degeneracy to diffusion sampling.

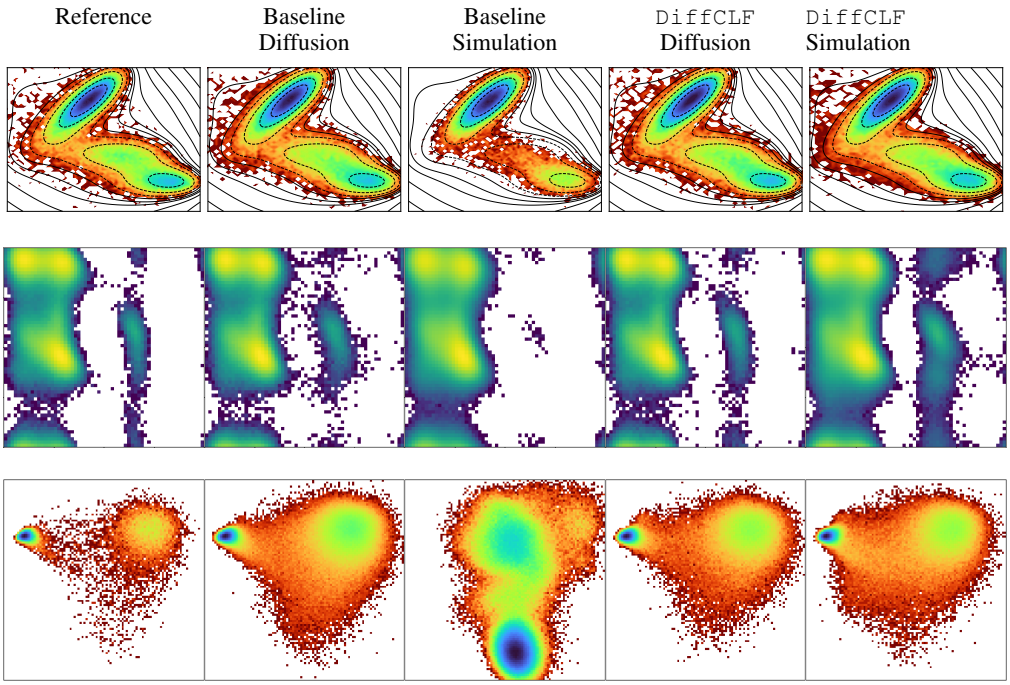


Figure 10: Comparison of samples drawn via *Diffusion sampling* (Diffusion) and *Langevin dynamics* (Simulation) by solely using the trained model $\mathrm{U}_{t=0}^\theta$. We compare DSM training only (Baseline) and joint training with the diffusive classification loss with 4 classes (`DiffCLF`). **(Top)**: Müller-Brown potential, visualization of the 2D histogram of the samples; **(Middle)**: Alanine Dipeptide, visualization of the 2D histogram of the torsion angles (x-axis: ϕ, y-axis: ψ); **(Bottom)**: Chignolin, visualization of the 2D histogram of the first two TIC coordinates.