[To be read before The Royal Statistical Society at the Society's 2023 annual conference held in Harrogate on Wednesday, September 6th, 2023, the President, Dr Andrew Garrett, in the Chair.]

From Denoising Diffusions to Denoising Markov Models

Joe Benton†
Department of Statistics, University of Oxford, Oxford, UK
Yuyang Shi
Department of Statistics, University of Oxford, Oxford, UK
Valentin De Bortoli
ENS, Paris, France
George Deligiannidis
Department of Statistics, University of Oxford, Oxford, UK
Arnaud Doucet
Department of Statistics, University of Oxford, Oxford, UK

Summary. Denoising diffusions are state-of-the-art generative models exhibiting remarkable empirical performance. They work by diffusing the data distribution into a Gaussian distribution and then learning to reverse this noising process to obtain synthetic datapoints. The denoising diffusion relies on approximations of the logarithmic derivatives of the noised data densities using score matching. Such models can also be used to perform approximate posterior simulation when one can only sample from the prior and likelihood. We propose a unifying framework generalising this approach to a wide class of spaces and leading to an original extension of score matching. We illustrate the resulting models on various applications.

Keywords: denoising diffusions, generative models, posterior simulation, score matching, unifying framework

1. Introduction

Given a set of samples from an unknown distribution $p_{\text{data}}(\mathbf{x})$, generative modelling is the task of producing further synthetic samples coming from approximately the same distribution. Over the past decade, a variety of techniques have been developed to tackle this problem, including autoregressive models (Oord et al., 2016), generative adversarial networks (Goodfellow et al., 2014), variational autoencoders (Kingma and Welling, 2014) and normalising flows (Rezende and Mohamed, 2015). These methods have had significant success in generating perceptually realistic samples from complex data distributions, such as text and image data (Brown et al., 2020; Dhariwal and Nichol, 2021). A major motivation for the development of generative models is that they can be easily

†*Address for correspondence:* Joe Benton, Department of Statistics, University of Oxford, 24-29 St Giles', Oxford, OX1 3LB, UK. E-mail: benton@stats.ox.ac.uk

extended for Bayesian inference. In a typical setting, we make an observation $\boldsymbol{\xi}^*$ based on underlying datapoint $\mathbf{x}$, for example a category label or partial observation of $\mathbf{x}$, and want to sample from the posterior distribution $p_{\text{data}}(\mathbf{x}|\boldsymbol{\xi}^*)$. We achieve this by learning a conditional generative model for $\mathbf{x}$ given any observation $\boldsymbol{\xi}$ based on samples from $p_{\text{data}}(\mathbf{x}, \boldsymbol{\xi})$. This approach is particularly useful in high-dimensional scenarios where traditional sampling methods, such as Markov chain Monte Carlo (MCMC) methods or approximate Bayesian computation (ABC), are typically infeasible.

Recently, denoising diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021) have emerged as effective generative models for high-dimensional data. They work by incrementally adding noise to the data to transform the data distribution into an easy-to-sample reference distribution, and then learning to invert the noising process, which is achieved using score matching (Hyvärinen, 2005). Their use for inference has recently seen an explosion of applications, including text-to-speech generation (Popov et al., 2021), image inpainting and super-resolution (Song et al., 2021; Saharia et al., 2022) and protein structure modelling (Trippe et al., 2023).

Most of the current methodology, theory and applications of denoising diffusion models are for diffusion processes on $\mathbb{R}^d$. However, many distributions of interest are defined on different spaces. Recently, De Bortoli et al. (2022) and Huang et al. (2022) have extended continuous-time methods and the analogy with score matching from $\mathbb{R}^d$ to general Riemannian manifolds in order to model data with strong geometric prior. Several diffusion methods have also been developed for discrete data, such as text, music or graph structures (Austin et al., 2021; Hoogeboom et al., 2021; Campbell et al., 2022; Sun et al., 2023). Here though, the relationships to score matching, as well as between these various methods and the Euclidean diffusion case, are less clear. All these recent extensions have been somewhat ad hoc, with training objectives needing to be re-derived for each new application.

The main contribution of this paper is to provide a unifying framework for such models, which we call *denoising Markov models*, or DMMs. We demonstrate how to construct and train a DMM for data in any state space satisfying mild regularity conditions. This yields a principled procedure for using these models for unconditional generation and inference on a wider class of spaces than previously considered. Additionally this general framework leads to a principled extension of score matching to general spaces. Finally, we demonstrate the application of our framework on examples in continuous space, discrete space, for Riemmanian manifolds and on the simplex.

2. Background

A denoising diffusion model is a generative model consisting of two stochastic processes. The *fixed* noising process takes a data point $\mathbf{x}_0$ drawn from a data distribution $q_0 := p_{\text{data}}$ on state space $\mathcal{X}$ and maps it stochastically to some $\mathbf{x}_T \in \mathcal{X}$. The *learned* generative process takes $\mathbf{x}_T \in \mathcal{X}$ drawn according to some initial distribution p_0 on $\mathcal{X}$ and maps it back stochastically to some $\mathbf{x}_0 \in \mathcal{X}$. Throughout, we denote the marginals of the noising and generative processes by $q_t(\mathbf{x})$ and $p_t(\mathbf{x})$ respectively for $t \in [0, T]$.

The basic idea is to pick a noising process so that $(q_t)_{t\geq 0}$ converges to some easy-to-sample-from distribution q_{ref}, which we then take to be p_0. We learn a generative

process which approximates the time-reversal of the noising process. Then, we can generate approximate samples from q_0 by sampling $\mathbf{x}_T \sim p_0$ and running the dynamics of the reverse process to produce a sample $\mathbf{x}_0 \sim p_T$, which should be close to q_0.

2.1. Continuous-time denoising diffusion models on $\mathbb{R}^d$

The framework for continuous-time diffusion models on $\mathbb{R}^d$ was first set out by Song et al. (2021). The noising process $(Y_t)_{t\in[0,T]}$ evolves according to the stochastic differential equation (SDE)

$$\mathrm{d}Y_t = b(Y_t, t)\mathrm{d}t + \mathrm{d}B_t, \qquad Y_0 = \mathbf{x}_0 \sim p_{\mathrm{data}}, \tag{1}$$

for some chosen function $b : \mathbb{R}^d \times [0,T] \to \mathbb{R}^d$, and standard Brownian motion B. With this set-up, the time-reversed process $X_t = Y_{T-t}$ can be simulated by initialising $X_0 = \mathbf{x}_T \sim q_T$ and running the SDE

$$\mathrm{d}X_t = \{-b(X_t, T-t) + \nabla_{\mathbf{x}} \log q_{T-t}(X_t)\}\mathrm{d}t + \mathrm{d}\hat{B}_t, \tag{2}$$

where $q_t(\mathbf{x}_t)$ denotes the marginals of the forward process and $\hat{B}$ is another standard Brownian motion (Anderson, 1982). We typically choose our forward process to be an Ornstein–Uhlenbeck process, i.e. $b(\mathbf{x}, t) = -\mathbf{x}/2$, for which $q_T \approx q_{\mathrm{ref}} := \mathcal{N}(0, I_d)$, the standard Gaussian distribution on $\mathbb{R}^d$, for large T.

To simulate the reverse process, we must approximate $\nabla_{\mathbf{x}} \log q_t(\mathbf{x})$. We do this by fixing a parametric family of functions $s_\theta(\mathbf{x}, t)$, and then choosing the parameters θ to minimise the *denoising score matching* objective

$$\mathcal{I}_{\mathrm{DSM}}(\theta) = \frac{1}{2}\int_0^T \mathbb{E}_{q_{0,t}(\mathbf{x}_0,\mathbf{x}_t)}\left[||\nabla_{\mathbf{x}} \log q_{t|0}(\mathbf{x}_t|\mathbf{x}_0) - s_\theta(\mathbf{x}_t, t)||^2\right] \mathrm{d}t, \tag{3}$$

where $q_{0,t}(\mathbf{x}_0, \mathbf{x}_t)$ and $q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$ denote the joint and conditional distributions of the SDE (1). The conditional is available in closed-form for the Ornstein–Uhlenbeck process. This is sensible since $\mathcal{I}_{\mathrm{DSM}}$ is minimised when $s_\theta(\mathbf{x}, t) = \nabla_{\mathbf{x}} \log q_t(\mathbf{x})$ for almost all $x \in \mathcal{X}$ and $t \in [0,T]$ (Song et al., 2021). If our score estimate were exact and $p_0 = q_T$, then we would have $p_t = q_{T-t}$ for all $t \in [0,T]$. In practice, we use a neural network to parameterise $s_\theta(\mathbf{x}, t)$ and use stochastic gradient descent to minimise $\mathcal{I}_{\mathrm{DSM}}(\theta)$.

Once we have a score estimate $s_\theta(\mathbf{x}, t)$, we compute approximate samples from the reverse process by running the approximate reverse process

$$\mathrm{d}X_t = \{-b(X_t, T-t) + s_\theta(X_t, T-t)\}\mathrm{d}t + \mathrm{d}\hat{B}_t \tag{4}$$

starting in $X_0 \sim p_0$ and setting $\mathbf{x}_0 = X_T$. In practice, we use suitable numerical integrators to simulate the approximate reverse process.

Alternatively, the objective $\mathcal{I}_{\mathrm{DSM}}$ can be derived from a lower bound on the model log-likelihood (also known as an Evidence Lower Bound, or ELBO) for $q_T(x)$, either using Girsanov's theorem and the chain rule for Kullback–Leibler divergences (Song et al., 2021), or by combining the Fokker–Planck equation and Feynman–Kac formula with Girsanov's theorem (Huang et al., 2021).

2.2. Diffusion models for inference

Denoising diffusions can also be used to sample approximately from a posterior $p_{\mathrm{data}}(\mathbf{x}|\boldsymbol{\xi}^*)$ when we only have access to samples from the joint distribution $p_{\mathrm{data}}(\mathbf{x}, \boldsymbol{\xi})$; see e.g. (Song

et al., 2021). We first draw a sample $(\mathbf{x}_0, \boldsymbol{\xi}_0) \sim p_{\text{data}}$, set $Y_0 = \mathbf{x}_0$ and let $(Y_t)_{t\in[0,T]}$ evolve according to Equation (1). If we condition on $\boldsymbol{\xi}_0$, then the process Y has marginals $q_t(\mathbf{x}_t|\boldsymbol{\xi}_0) = \int q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)p_{\text{data}}(\mathbf{x}_0|\boldsymbol{\xi}_0)\mathrm{d}\mathbf{x}_0$, where $q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$ is the transition kernel of the forward diffusion in Equation (1). So, the time-reversed process $X_t = Y_{T-t}$ conditioned on $\boldsymbol{\xi}_0$ can be simulated by initialising $X_0 \sim q_T(\cdot|\boldsymbol{\xi}_0)$ and running the SDE

$$\mathrm{d}X_t = \{-b(X_t, T-t) + \nabla_{\mathbf{x}} \log q_{T-t}(X_t \mid \boldsymbol{\xi}_0)\}\mathrm{d}t + \mathrm{d}\hat{B}_t. \tag{5}$$

If we have $q_T(\cdot|\boldsymbol{\xi}) \approx q_{\text{ref}}$ for all $\boldsymbol{\xi}$ and an approximation $s_\theta(\mathbf{x}, \boldsymbol{\xi}, t)$ to $\nabla_{\mathbf{x}} \log q_t(\mathbf{x}|\boldsymbol{\xi})$, we can obtain approximate samples from $q_0(\cdot|\boldsymbol{\xi}^*) = p_{\text{data}}(\cdot|\boldsymbol{\xi}^*)$ for any given $\boldsymbol{\xi}^*$ by initialising $X_0 \sim p_0 := q_{\text{ref}}$, simulating the reverse dynamics in Equation (5) with $\nabla_{\mathbf{x}} \log q_{T-t}(X_t|\boldsymbol{\xi}_0)$ replaced by $s_\theta(X_t, \boldsymbol{\xi}^*, T-t)$, and setting $\mathbf{x}_0 = X_T$. To learn $s_\theta(\mathbf{x}, \boldsymbol{\xi}, t)$, we minimise

$$\mathcal{I}_{\text{DSM}}(\theta) = \frac{1}{2}\int_0^T \mathbb{E}_{q(\mathbf{x}_0, \mathbf{x}_t, \boldsymbol{\xi}_0)}\left[||\nabla_{\mathbf{x}} \log q_{t|0}(\mathbf{x}_t|\mathbf{x}_0) - s_\theta(\mathbf{x}_t, \boldsymbol{\xi}_0, t)||^2\right] \mathrm{d}t,$$

where we denote $q(\mathbf{x}_0, \mathbf{x}_t, \boldsymbol{\xi}_0) = p_{\text{data}}(\mathbf{x}_0, \boldsymbol{\xi}_0)q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$. This objective is minimised when $s_\theta(\mathbf{x}, \boldsymbol{\xi}, t) = \nabla_{\mathbf{x}} \log q_t(\mathbf{x}|\boldsymbol{\xi})$ for almost all $x \in \mathcal{X}$ and $t \in [0, T]$ (Song et al., 2021).

2.3. Score matching

The objective $\mathcal{I}_{\text{DSM}}$ defined in Equation (3) can also be interpreted as a score matching objective. Score matching was introduced as a method for fitting unnormalised probability distributions defined on $\mathbb{R}^d$ by Hyvärinen (2005). It approximates a distribution $q_0(\mathbf{x})$ with a distribution of the form $p(\mathbf{x}; \theta) = q(\mathbf{x}; \theta)/Z(\theta)$ by minimising

$$\mathcal{J}(\theta) = \frac{1}{2}\mathbb{E}_{q_0(\mathbf{x})}\left[||\nabla_{\mathbf{x}} \log q_0(\mathbf{x}) - \nabla_{\mathbf{x}} \log q(\mathbf{x}; \theta)||^2\right],$$

known as an *explicit score matching* loss. This objective is intractable since it depends on $\nabla_{\mathbf{x}} \log q_0(\mathbf{x})$, but there are methods for rewriting it in an equivalent tractable form, including implicit and denoising score matching (Hyvärinen, 2005; Vincent, 2011). Equation (3), which corresponds to denoising score matching, can also be written in explicit, implicit or sliced score matching form (Huang et al., 2021).

3. A general framework for denoising Markov models

In this section, we set out a general framework for DMMs. First, we explain how to construct a DMM on an arbitrary state space with a forward noising process Y and backward generative process X. Second, we derive an expression for the model likelihood in terms of an expectation over an auxiliary process Z, defined in terms of X and running forward in time. Third, we derive an ELBO by using Girsanov's theorem to relate the expectation over Z to one over Y. Finally, we show how this ELBO can be used to get a tractable training objective. Our argument follows a similar structure to Huang et al. (2021), but we work in terms of generic Markov generators, rather than specific operators corresponding to diffusions on $\mathbb{R}^d$, and so require generalisations of the stochastic process results therein. For simplicity, we present the framework for unconditional generation and then explain how to adapt it for inference.

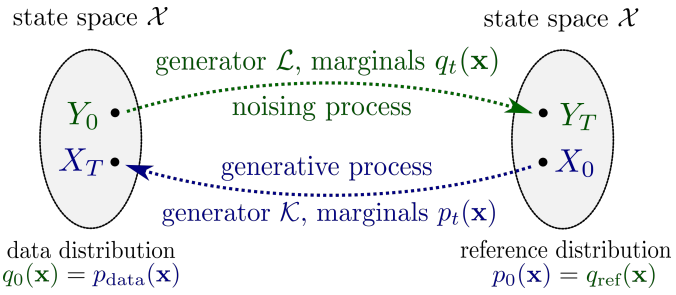

Fig. 1. Diagram of notation.

3.1. Notation and set-up

Our data is assumed to be distributed according to p_{data} on a state space $\mathcal{X}$. We assume only that $\mathcal{X}$ comes with some reference measure ν, with respect to which all probability densities will be defined, and satisfies some regularity conditions given in Appendix B.1. This includes $\mathbb{R}^d$, discrete spaces and Riemannian manifolds (with or without boundary).

Our DMM consists of a noising process $(Y_t)_{t\in[0,T]}$ and a generative process $(X_t)_{t\in[0,T]}$, which are Markov processes. We consider Y fixed and learn X to approximate the reverse of Y. Initially, we must fix a class of processes to which X and Y belong and within which we will optimise X. The particular class and parameterisation we choose will necessarily depend on $\mathcal{X}$, but a typical choice for $\mathcal{X} = \mathbb{R}^d$ would be a diffusion (see Example 1), while a typical choice when $\mathcal{X}$ is a finite discrete space may be a continuous-time Markov chain (CTMC) (see Example 2). Our notation is depicted in Fig. 1.

As X and Y are not necessarily time-homogeneous, it is helpful to define the extended processes $\overline{X}$ and $\overline{Y}$ by for example setting $X_t = X_T$ for $t \geq T$ and letting $\overline{X} = (X_t, t)_{t\geq 0}$. Then $\overline{X}$, $\overline{Y}$ are time-homogeneous Markov chains on the extended space $\mathcal{S} := \mathcal{X} \times [0, \infty)$.

In general, it is most convenient to define X and Y via the generators of $\overline{X}$ and $\overline{Y}$, which we denote by $\mathcal{K}$ and $\mathcal{L}$ respectively. Informally, the generator of a Markov process $\overline{W}$ with state space $\mathcal{S}$ is an operator $\mathcal{A}$ which acts on a subset $\mathcal{D}(\mathcal{A})$ of the space of functions $f : \mathcal{S} \to \mathbb{R}$ and satisfies $\mathcal{A}f = \lim_{s\to 0}(P_s f - f)/s$, where $(P_s)_{s\geq 0}$ is the transition semigroup associated to $\overline{W}$ and $P_s f(x) = \mathbb{E}\left[f(X_s)|X_0 = x\right]$. For a more formal definition, see Appendix A.1.

We denote the time marginals of the processes X, Y by $p_t(\mathbf{x})$, $q_t(\mathbf{x})$ respectively. We make some smoothness assumptions on p, in Appendix B.2, and assume that $\mathcal{K}$, $\mathcal{L}$ satisfy some regularity conditions, in Appendix B.3. Our assumptions hold for standard models in the literature (Euclidean diffusions, CTMCs and manifold diffusions; see Appendix F), plus some that are not covered previously, such as degenerate diffusions. For infinite dimensional spaces, the assumptions of Appendix B.1 may fail and more care is needed.

One consequence of our assumptions is that the operator $\mathcal{K}$ decomposes as $\mathcal{K} = \partial_t + \hat{\mathcal{K}}$, where $\hat{\mathcal{K}}$ operates only on the spatial variables of a function f. We can therefore view $\hat{\mathcal{K}}$ as an operator on functions from $\mathcal{X}$, rather than on functions from $\mathcal{S}$, and we denote by $\hat{\mathcal{K}}^*$ the adjoint of $\hat{\mathcal{K}}$ acting on functions on $\mathcal{X}$ (see Appendix A.2).

Example 1 (Euclidean Diffusion). *If X and Y are diffusions on $\mathbb{R}^d$ given by the SDEs $\mathrm{d}X_t = \mu(X_t, t)\mathrm{d}t + \mathrm{d}\hat{B}_t$ and $\mathrm{d}Y_t = b(Y_t, t)\mathrm{d}t + \mathrm{d}B_t$, where B and $\hat{B}$ are Brownian*

motions, then the corresponding generators are $\mathcal{K} = \partial_t + \mu \cdot \nabla + \frac{1}{2}\Delta$ *and* $\mathcal{L} = \partial_t + b \cdot \nabla + \frac{1}{2}\Delta$*, where* $\Delta = \sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2}$ *denotes the Laplacian. We then have* $\hat{\mathcal{K}}^* = -\mu \cdot \nabla - (\nabla \cdot \mu) + \frac{1}{2}\Delta$ *using integration by parts.*

Example 2 (Discrete Space CTMC). *If* X *and* Y *are CTMCs, then* $\mathcal{K} = \partial_t + A$ *and* $\mathcal{L} = \partial_t + B$*, where* A *and* B *are the time-dependent generator matrices of* X *and* Y*. In this case,* $\hat{\mathcal{K}}^* = A^T$*, the transpose of* A*.*

3.2. An expression for the model likelihood

We now derive an expression for the model likelihood $p_T(\mathbf{x})$. First, under our assumptions, a generalised form of the Fokker–Planck equation, stated precisely in Appendix C, implies that $\partial_t p = \hat{\mathcal{K}}^* p$ for ν-almost every $\mathbf{x} \in \mathcal{X}$. Typically, the adjoint operator $\hat{\mathcal{K}}^*$ resembles the generator of another process in the same class as X and Y. We formalise this idea by making the following assumption.

Assumption 1. *Let* $v(\mathbf{x}, t) = p_{T-t}(\mathbf{x})$*. Then we can write the equation* $\partial_t p = \hat{\mathcal{K}}^* p$ *in the form* $\mathcal{M}v + cv = 0$ *for some function* $c : \mathcal{S} \to \mathbb{R}$*, where* $\mathcal{M}$ *is the generator of another auxiliary Feller process* $\overline{Z} = (Z_t, t)_{t \geq 0}$ *on* $\mathcal{S}$*.*

Example 3 (Euclidean Diffusion). *For Euclidean diffusions, the Fokker–Planck equation can be written as* $\partial_t v = \mu \cdot \nabla v + (\nabla \cdot \mu)v - \frac{1}{2}\Delta v$*. Assumption 1 is satisfied with* $c = -(\nabla \cdot \mu)$ *and* $\mathcal{M} = \partial_t - \mu \cdot \nabla + \frac{1}{2}\Delta$*, noting that* $\mathcal{M}$ *is the generator of the diffusion process* Z *defined by* $\mathrm{d}Z_t = -\mu(Z_t, T - t)\mathrm{d}t + \mathrm{d}B'_t$*, where* B' *is a Brownian motion.*

Example 4 (Discrete Space CTMC). *In the CTMC case, if* $c_{\mathbf{x}} = \sum_{\mathbf{y} \in \mathcal{X}} A_{\mathbf{yx}}$*, and* $D_{\mathbf{xy}} = A_{\mathbf{yx}} - c_{\mathbf{x}} \mathbb{1}_{\mathbf{x}=\mathbf{y}}$*, then* $\mathcal{M} = \partial_t + D$ *is the generator of a CTMC and Assumption 1 is satisfied. Here* c *has a natural interpretation as a "discrete divergence".*

In general, we make two smoothness assumptions on c and v, given in Appendix B.4.

Given the Fokker–Planck equation and Assumption 1, we apply a generalised form of the Feynman–Kac Theorem (see Appendix C) to $\overline{Z}$ and v to get the following expression for the model likelihood, which generalises that of Huang et al. (2021):

$$p_T(\mathbf{x}) = v(\mathbf{x}, 0) = \mathbb{E}\left[p_0(Z_T) \exp\left\{ \int_0^T c(Z_s, s)\, \mathrm{d}s \right\} \;\middle|\; Z_0 = \mathbf{x} \right]. \tag{6}$$

This gives an expression in terms of an expectation over the auxiliary process Z. We next make this tractable by converting it into an expectation over Y.

3.3. Deriving a tractable lower bound on the model log-likelihood

We would like to train our model by finding a reverse process X which maximises the likelihood in Equation (6). Unfortunately this expression is intractable, but we can find a tractable lower bound for $\log p_T(\mathbf{x})$ which can then be used as a surrogate objective.

By taking logarithms in Equation (6) and applying Jensen's inequality, we get

$$\log p_T(\mathbf{x}) \geq \mathbb{E}_{\mathbb{Q}}\left[\log \frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\mathbb{Q}} + \log p_0(Y_T) + \int_0^T c(Y_s, s)\, \mathrm{d}s \;\middle|\; Y_0 = \mathbf{x} \right] =: \mathcal{E}^{\infty} \tag{7}$$

where $\mathbb{P}$ and $\mathbb{Q}$ are the path measures of the processes $\overline{Z}$ and $\overline{Y}$ respectively and $\frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\mathbb{Q}}$ denotes the Radon-Nikodym derivative.

To write $\mathcal{E}^\infty$ in a tractable form we need to evaluate $\log \frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\mathbb{Q}}$, which we do using a generalisation of Girsanov's theorem. To apply this result, we require that the generators of the auxiliary process and the noising process are related in the following way.

ASSUMPTION 2. *There is a bounded measurable function $\beta : \mathcal{S} \to (0, \infty)$ such that $\beta^{-1}\mathcal{M}f = \mathcal{L}(\beta^{-1}f) - f\mathcal{L}(\beta^{-1})$ for all $f : S \to \mathbb{R}$ such that $f \in \mathcal{D}(\mathcal{M})$ and $\beta^{-1}f \in \mathcal{D}(\mathcal{L})$.*

Since $\mathcal{M}$ is defined in terms of $\mathcal{K}$, we think of Assumption 2 as forcing a particular parameterisation of the generative process in terms of β. In general, not every generative process in the same class as $\mathcal{L}$ will have such a parameterisation. However, the true time-reversal of $\mathcal{L}$ can always be parameterised in this way with $\beta(\mathbf{x}, t) = p_t(\mathbf{x})$, so this parameterisation is sufficient to capture the optimal generative process. In addition, the objective in Theorem 1 below can often be interpreted and used for a much broader set of generative processes than those which satisfy Assumption 2.

Under Assumption 2, along with a further technical assumption given in Appendix B.5, we may apply a generalised form of Girsanov's Theorem (see Appendix C, and take $\alpha = \beta^{-1}$ in Theorem 6) and Dynkin's formula (see Appendix A.1) to get

$$\log \frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\mathbb{Q}} = \int_0^T \left\{ -\mathcal{L} \log \beta(Y_s, s) - \beta(Y_s, s)\mathcal{L}(\beta^{-1})(Y_s, s) \right\} \mathrm{d}s + \mathbb{Q}\text{-martingale}.$$

In addition, we get that $c = \beta\mathcal{L}(\beta^{-1}) - v^{-1}\beta\mathcal{L}(\beta^{-1}v)$ by combining Assumption 2 with $f = v$ and Assumption 1. This allows us to rewrite the ELBO from Equation (7) as

$$\mathcal{E}^\infty = \mathbb{E}_{\mathbb{Q}} \left[\log p_0(Y_T) - \int_0^T \left\{ \frac{\mathcal{L}(\beta^{-1}v)}{\beta^{-1}v} + \mathcal{L} \log \beta \right\} \mathrm{d}s \;\middle|\; Y_0 = \mathbf{x} \right].$$

The final step required to get a tractable expression for $\mathcal{E}^\infty$ is to remove the function v from this expression. For this, we use the following lemma (see Appendix D).

LEMMA 1. *Let the generator $\mathcal{L}$ and the functions β and c be as above. Then, we have $v^{-1}\beta\mathcal{L}(\beta^{-1}v) + \mathcal{L} \log \beta = \beta^{-1}\hat{\mathcal{L}}^*\beta + \hat{\mathcal{L}} \log \beta$.*

THEOREM 1. *For DMMs as in Section 3.1–3.3, the log-likelihood is lower bounded by*

$$\mathcal{E}^\infty = \mathbb{E}_{\mathbb{Q}} \Big[\log p_0(Y_T) \Big| Y_0 = \mathbf{x} \Big] - \int_0^T \mathbb{E}_{\mathbb{Q}} \left[\frac{\hat{\mathcal{L}}^*\beta}{\beta} + \hat{\mathcal{L}} \log \beta \;\middle|\; Y_0 = \mathbf{x} \right] \mathrm{d}s. \tag{8}$$

This result extends the corresponding expression for $\mathbb{R}^d$ in Huang et al. (2021). We see the ELBO consists of a term representing the log-likelihood under the reference distribution and an implicit score matching term arising from the change in measure.

3.4. Finding suitable training objectives

Based on Theorem 1, we fit our generative model by maximising the expectation of $\mathcal{E}^\infty$ with respect to p_{data}. This is equivalent to minimising the objective

$$\mathcal{I}_{\text{ISM}}(\beta) = \int_0^T \mathbb{E}_{q_t(\mathbf{x}_t)} \left[\frac{\hat{\mathcal{L}}^*\beta(\mathbf{x}_t, t)}{\beta(\mathbf{x}_t, t)} + \hat{\mathcal{L}} \log \beta(\mathbf{x}_t, t) \right] \mathrm{d}t, \tag{9}$$

which we call the *implicit score matching* objective, since it can be interpreted as an extension of implicit score matching from $\mathbb{R}^d$ (see Section 4 below for more intuition).

Since q_t and $\hat{\mathcal{L}}$ are determined by the noising process, which is known and assumed easy to sample from, $\mathcal{I}_{\mathrm{ISM}}(\beta)$ and its gradient with respect to β can be estimated in an unbiased fashion. Since β parameterises $\mathcal{M}$ via Assumption 2, and thus $\mathcal{K}$ through Assumption 1, minimising $\mathcal{I}_{\mathrm{ISM}}(\beta)$ over β is equivalent to learning the generative process.

We also have an equivalent *denoising score matching objective* (see Appendix E),

$$\mathcal{I}_{\mathrm{DSM}}(\beta) = \int_0^T \mathbb{E}_{q_{0,t}(\mathbf{x}_0,\mathbf{x}_t)} \left[\frac{\mathcal{L}(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\cdot))(\mathbf{x}_t,t)}{q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)/\beta(\mathbf{x}_t,t)} - \mathcal{L}\log(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\cdot))(\mathbf{x}_t,t) \right] \mathrm{d}t. \tag{10}$$

Both objectives are minimised when $\beta(\mathbf{x},t) \propto q_t(\mathbf{x})$, as shown in Proposition 1. $\mathcal{I}_{\mathrm{DSM}}(\beta)$ can be interpreted as quantifying the difference between $q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$ and $\beta(\mathbf{x}_t,t)$ via the score matching operator $\Phi(f) = f^{-1}\mathcal{L}f - \mathcal{L}\log f$ introduced in Section 4 below. These objectives also generalise the following previously studied instances of diffusion models. For all derivations and remarks on the choice of parameterisation, see Appendix F.

Example 5 (Euclidean Diffusion). *In the setting of Example 1, Assumption 2 reduces to* $\nabla\log\beta = b + \mu$*, and we have* $f^{-1}\mathcal{L}f - \mathcal{L}\log f = \frac{1}{2}\left\|\nabla\log f\right\|^2$*. If we substitute* $s_\theta(\mathbf{x}_t,t) = \nabla\log\beta(\mathbf{x}_t,t)$*,* $\mathcal{I}_{\mathrm{DSM}}(\beta)$ *defined in Equation (10) reduces to Equation (3) and the reverse process is parameterised as in Equation (4). We thus recover the results of Song et al. (2021) and Huang et al. (2021).*

Example 6 (Discrete Space CTMC). *In the setting of Example 2, Assumption 2 reduces to* $A_{\mathbf{yx}} = \frac{\beta(\mathbf{x},t)}{\beta(\mathbf{y},t)}B_{\mathbf{xy}}$ *for all* $\mathbf{x} \neq \mathbf{y}$*. We may rewrite* $\mathcal{I}_{\mathrm{ISM}}$ *in terms of* A *to recover the objective of Campbell et al. (2022),*

$$\mathcal{I}_{\mathrm{ISM}}(A) = \int_0^T \mathbb{E}_{q_t(\mathbf{x}_t)} \Bigg[-A_{\mathbf{x}_t\mathbf{x}_t} - \sum_{\mathbf{y}\neq\mathbf{x}_t} B_{\mathbf{x}_t\mathbf{y}} \log A_{\mathbf{y}\mathbf{x}_t} \Bigg] \mathrm{d}t + \mathit{const}.$$

Example 7 (Riemannian Manifolds). *If* $\mathcal{X}$ *is a Riemannian manifold and we take* $\mathcal{K} = \partial_t + \mu\cdot\nabla + \frac{1}{2}\Delta$*,* $\mathcal{L} = \partial_t + b\cdot\nabla + \frac{1}{2}\Delta$ *where* Δ *is the Laplace–Beltrami operator associated to* $\mathcal{X}$*, and perform the reparameterisation* $s_\theta(\mathbf{x}_t,t) = \nabla\log\beta(\mathbf{x}_t,t)$*, then we recover the framework for training diffusion models on Riemannian manifolds given in De Bortoli et al. (2022) and Huang et al. (2022).*

3.5. Inference

To use DMMs for inference, we follow a similar procedure to Section 2.2. To noise a sample $(\mathbf{x}_0, \boldsymbol{\xi}_0) \sim p_{\mathrm{data}}$, we set $Y_0 = \mathbf{x}_0$ and let Y evolve according to $\mathcal{L}$. To generate $\mathbf{x}_0$ conditioned on an observation $\boldsymbol{\xi}^*$, we use a generative process $X^{\boldsymbol{\xi}^*}$ conditioned on $\boldsymbol{\xi}^*$. We parameterise $X^{\boldsymbol{\xi}^*}$ in terms of a function $\beta(\mathbf{x}_t, \boldsymbol{\xi}^*, t)$ which now takes $\boldsymbol{\xi}^*$ as an input.

We aim to learn $X^{\boldsymbol{\xi}^*}$ to approximate the time-reversal of Y conditioned on $\boldsymbol{\xi}^*$. The following extension of Theorem 1 (proved in Appendix D) gives us a way to do this.

THEOREM 2. *With the above set-up, minimising the objective*

$$\mathcal{I}_{\mathrm{DSM}}(\beta) = \int_0^T \mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t,\boldsymbol{\xi}_0)}\left[\frac{\mathcal{L}(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\boldsymbol{\xi}_0,\cdot))(\mathbf{x}_t,t)}{q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)/\beta(\mathbf{x}_t,\boldsymbol{\xi}_0,t)} - \mathcal{L}\log(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\boldsymbol{\xi}_0,\cdot))(\mathbf{x}_t,t)\right]\mathrm{d}t$$

is equivalent to maximising a lower bound on the expected model log-likelihood.

Theorem 2 suggests that we may train conditional DMMs by maximising the objective $\mathcal{I}_{\mathrm{DSM}}(\beta)$ (or the equivalent $\mathcal{I}_{\mathrm{ISM}}(\beta)$ objective). Since $q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$ is known, we may do this by calculating an empirical estimate for $\mathcal{I}_{\mathrm{DSM}}(\beta)$ based on samples $(\mathbf{x}_0, \boldsymbol{\xi}_0)$ drawn from p_{data} and minimising over β. Then, we generate samples from $p_{\mathrm{data}}(\mathbf{x}_0|\boldsymbol{\xi}^*)$ by initialising $X_0^{\boldsymbol{\xi}^*} \sim p_0$, simulating the reverse process with generator $\mathcal{K}$ parameterised by $\beta = \beta(\cdot, \boldsymbol{\xi}^*, \cdot)$, and setting $\mathbf{x}_0 = X_T^{\boldsymbol{\xi}^*}$.

4. Score matching on general state-spaces

When X and Y are Euclidean diffusions, the objective $\mathcal{I}_{\mathrm{DSM}}(\beta)$ in Equation (10) becomes the score matching objective in Equation (3). Similarly, the objective $\mathcal{I}_{\mathrm{ISM}}(\beta)$ from Equation (9) reduces to the implicit score matching objective introduced by Hyvärinen (2005). This suggests we can view Equations (9) and (10) as generalisations of score matching objectives to arbitrary state spaces.

Given state space $\mathcal{X}$ on which we have a Markov process generator $\mathcal{L}$ and an unknown distribution $q_0(\mathbf{x})$ we wish to approximate, the corresponding *generalised implicit score matching* method learns an approximation $\varphi(\mathbf{x})$ to $q_0(\mathbf{x})$ by minimising

$$\mathcal{J}_{\mathrm{ISM}}(\varphi) = \mathbb{E}_{q_0(\mathbf{x})}\left[\frac{\hat{\mathcal{L}}^*\varphi(\mathbf{x})}{\varphi(\mathbf{x})} + \hat{\mathcal{L}}\log\varphi(\mathbf{x})\right].$$

We can show that $\mathcal{J}_{\mathrm{ISM}}$ is equivalent to the *generalised explicit score matching objective*

$$\mathcal{J}_{\mathrm{ESM}}(\varphi) = \mathbb{E}_{q_0(\mathbf{x})}\left[\frac{\mathcal{L}(q_0/\varphi)(\mathbf{x})}{(q_0(\mathbf{x})/\varphi(\mathbf{x}))} - \mathcal{L}\log(q_0/\varphi)(\mathbf{x})\right].$$

In addition, we define the corresponding *generalised denoising score matching* method, which learns an approximation $\varphi_\tau(\mathbf{x}_\tau)$ to the noised distribution $q_\tau(\mathbf{x}_\tau)$, formed by sampling $\mathbf{x}_0 \sim q_0(\cdot)$ and $\mathbf{x}_\tau \sim q_{\tau|0}(\,\cdot\,|\mathbf{x}_0)$, where $q_{\tau|0}$ is the transition probability associated to $\mathcal{L}$ run for time τ. It does this by minimising the objective

$$\mathcal{J}_{\mathrm{DSM}}(\varphi_\tau) = \mathbb{E}_{q_{0,\tau}(\mathbf{x}_0,\mathbf{x}_\tau)}\left[\frac{\mathcal{L}(q_{\tau|0}(\cdot|\mathbf{x}_0)/\varphi_\tau(\cdot))(\mathbf{x}_\tau)}{q_{\tau|0}(\mathbf{x}_\tau|\mathbf{x}_0)/\varphi_\tau(\mathbf{x}_\tau)} - \mathcal{L}\log(q_{\tau|0}(\cdot|\mathbf{x}_0)/\varphi_\tau(\cdot))(\mathbf{x}_\tau)\right].$$

$\mathcal{J}_{\mathrm{DSM}}$ is equivalent to both $\mathcal{J}_{\mathrm{ISM}}$ and $\mathcal{J}_{\mathrm{ESM}}$ when used to learn the smoothed distribution $q_\tau(\mathbf{x}_\tau)$ (see Appendix E). All three objectives extend the corresponding score matching objectives introduced for $\mathbb{R}^d$ by Hyvärinen (2005) and Vincent (2011). They also coincide with the extension of score matching for Riemannian manifolds of Mardia et al. (2016).

To illustrate further intuitions behind our objective functions, we define the *score matching operator* $\Phi(f) = f^{-1}\mathcal{L}f - \mathcal{L}\log f$. Note that the time component of Φ cancels,

so we can view it as an operator on $\mathcal{X}$. With this notation, the generalised explicit score matching objective becomes $\mathcal{J}_{\text{ESM}}(\varphi) = \mathbb{E}_{q_0(\mathbf{x})}[\Phi(q_0/\varphi)(\mathbf{x})]$. For Euclidean diffusions, $\Phi(f) = \frac{1}{2}||\nabla \log f||^2$ (see Example 5). In the general case, we view $\Phi(f)$ as measuring the magnitude of a logarithmic gradient of f. We interpret the objectives $\mathcal{J}_{\text{DSM}}$ and $\mathcal{J}_{\text{ESM}}$ as trying to fit φ to q_0 by minimising this logarithmic gradient of the ratio q_0/φ.

PROPOSITION 1. *Let Y be a Feller process with semigroup operators $(Q_t)_{t\geq 0}$, generator $\mathcal{L}$ and associated score matching operator Φ. Then:*

(a) $\Phi(f) \geq 0$ for all f in the domain of Φ, with equality if f is constant;
(b) for any probability measures π_1, π_2 on $\mathcal{X}$ and $t \geq 0$,

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{KL}(\pi_1 Q_t||\pi_2 Q_t) = -\mathbb{E}_{\pi_1 Q_t}\left[\Phi\left(\frac{\mathrm{d}(\pi_1 Q_t)}{\mathrm{d}(\pi_2 Q_t)}\right)\right],$$

where $\mathrm{KL}(\pi_1 Q_t||\pi_2 Q_t)$ denotes the Kullback–Leibler divergence between $\pi_1 Q_t$, $\pi_2 Q_t$.

Proposition 1(a) shows that Φ is always non-negative, so $\mathcal{J}_{\text{ESM}}$ is minimised if $\varphi(\mathbf{x}) \propto q_0(\mathbf{x})$. Thus minimising any of our generalised score matching objectives should typically correspond to learning an approximation to q_0. Note though that if Q_t is not ergodic and π_1, π_2 are different invariant distributions of Q_t then Proposition 1(b) implies that $\Phi(\mathrm{d}\pi_1/\mathrm{d}\pi_2) = 0$ π_1-a.e., even though $\mathrm{d}\pi_1/\mathrm{d}\pi_2$ is not constant. This suggests that generalised score matching may fail if the noising process is not ergodic. Proposition 1(b) was proved for score matching on $\mathbb{R}^d$ by Lyu (2009). It suggests we can interpret score matching as finding an approximation φ which minimises the decrease in KL divergence between q_0 and φ caused by adding an infinitesimal amount of noise to both according to $\mathcal{L}$.

Our generalised score matching methods give a principled way to extend score matching to fit unnormalised probability distributions on arbitrary spaces. Other extensions of score matching have been explored, including to arbitrary sub-domains of $\mathbb{R}^d$ (Yu et al., 2022), ratio matching (Hyvärinen, 2007) and marginalisation with generalised score matching (Lyu, 2009). However, these methods lack the generality of our framework and do not respect the intuition coming from $\mathbb{R}^d$ that Proposition 1(b) should hold. There are also many other density estimation methods that seek to learn ratios of density functions, including noise-contrastive estimation, which also approximates score matching under certain conditions (Gutmann and Hirayama, 2011).

5. Relationship to discrete time models

Denoising diffusion models were originally introduced in discrete time by Sohl-Dickstein et al. (2015). In this setting, the noising and generative processes are Markov chains $\mathbf{x}_{0:T} = (\mathbf{x}_{t_k})_{k=0}^N$ observed at a sequence of times $0 = t_0 < t_1 < \cdots < t_N = T$, with fixed forwards transition kernel $\tilde{q}(\mathbf{x}_{t_k}|\mathbf{x}_{t_{k-1}})$ and learned backwards kernel $\tilde{p}_\theta(\mathbf{x}_{t_{k-1}}|\mathbf{x}_{t_k})$. To fit discrete time diffusion models, Sohl-Dickstein et al. (2015) minimise the following Kullback–Leibler divergence with respect to θ:

$$\mathrm{KL}(\tilde{q}(\mathbf{x}_{0:T})||\tilde{p}_\theta(\mathbf{x}_{0:T})) = \sum_{k=1}^{N} \mathbb{E}_{\tilde{q}(\mathbf{x}_{t_{k-1}}, \mathbf{x}_{t_k})}\left[\log \frac{\tilde{q}(\mathbf{x}_{t_k}|\mathbf{x}_{t_{k-1}})}{\tilde{p}_\theta(\mathbf{x}_{t_{k-1}}|\mathbf{x}_{t_k})}\right] + \textit{const.} \tag{11}$$

Given any DMM with generators $K, \mathcal{L}$ and marginals p_t, q_t as in Section 3, we define its *natural discretisation* to be the discrete-time model with $\tilde{q}(\mathbf{x}_{t_k}|\mathbf{x}_{t_{k-1}}) = q_{t_k|t_{k-1}}(\mathbf{x}_{t_k}|\mathbf{x}_{t_{k-1}})$ and $\tilde{p}_\theta(\mathbf{x}_{t_{k-1}}|\mathbf{x}_{t_k}) = p_{T-t_{k-1}|T-t_k}(\mathbf{x}_{t_{k-1}}|\mathbf{x}_{t_k})$. Then, the Kullback–Leibler divergence (11) for the natural discretisation can be viewed as a first-order approximation to $\mathcal{I}_{\mathrm{ISM}}$ for the continuous-time model.

LEMMA 2. *Suppose X, Y are fixed generative and noising processes with marginals p, q as in Section 3, and suppose that they are related as in Assumptions 1 and 2 for some sufficiently regular function β. Then for any $0 < s < t < T$ with $\gamma = t - s$,*

$$\gamma\, \mathbb{E}_{q_s(\mathbf{x}_s)}\left[\frac{\hat{\mathcal{L}}^*\beta}{\beta} + \hat{\mathcal{L}}\log\beta\right] = \mathbb{E}_{q_{s,t}(\mathbf{x}_s,\mathbf{x}_t)}\left[\log\frac{q_{t|s}(\mathbf{x}_t|\mathbf{x}_s)}{p_{T-s|T-t}(\mathbf{x}_s|\mathbf{x}_t)}\right] + o(\gamma).$$

Applying this lemma on each interval $[t_k, t_{k+1}]$, we get the following theorem.

THEOREM 3. *For any DMM, the objective (11) for its natural discretisation is equivalent to the natural discretisation of $\mathcal{I}_{\mathrm{ISM}}$ to first order in $\overline{\gamma} = \max_{k=0,\dots,N-1}|t_{k+1} - t_k|$.*

This theorem generalises to arbitrary state spaces a result of Ho et al. (2020), which demonstrated the equivalence of minimizing (11) and the score matching objective for Euclidean state spaces. For the proofs of Lemma 2 and Theorem 3, see Appendix H.

Lemma 2 also implies a general equivalence between one-step denoising autoencoders and score matching. Vincent (2011) discussed this equivalence for autoencoders using Gaussian noise in $\mathbb{R}^d$, but our methods allow us to extend this correspondence to arbitrary state spaces and noising processes. For more details, see Appendix I.

6. Experiments

We now present experiments demonstrating DMMs on several tasks and data spaces, for unconditional generation and conditional simulation. All details are in Appendix J.

6.1. Inference on $\mathbb{R}^d$ using diffusion processes

First, we use diffusion processes in $\mathbb{R}^d$ to perform approximate Bayesian inference for real-valued parameters. We consider $p_{\mathrm{data}}(\boldsymbol{\xi}|\mathbf{x}) = \prod_{i=1}^N p_{\mathrm{data}}(\xi_i|\mathbf{x})$, where $p_{\mathrm{data}}(\xi_i|\mathbf{x})$ is the g-and-k distribution with parameters $\mathbf{x} = (A, B, g, k)$ and $d = 4$, and we let $p_{\mathrm{data}}(\mathbf{x})$ be uniform on $[0, 10]^4$. The g-and-k distribution is a 4-parameter distribution in which A, B, g, k control the location, scale, skewness and kurtosis respectively.

We fix our noising process to be an Ornstein–Uhlenbeck process, and parameterise our reverse process as in Example 5, with $s_\theta(\mathbf{x}, \boldsymbol{\xi}, t)$ being given by a fully connected neural network. To train the model, we sample $(\mathbf{x}_0, \boldsymbol{\xi}_0) \sim p_{\mathrm{data}}$ and minimise the denoising score matching objective from Section 3.5 via stochastic gradient descent on θ.

To test our model, we first consider the case where there are a true set of underlying parameters $\mathbf{x}_{\mathrm{true}} = (3, 1, 2, 0.5)$. We generate an observation $\boldsymbol{\xi}_0 \sim p_{\mathrm{data}}(\boldsymbol{\xi}_0|\mathbf{x}_{\mathrm{true}})$ with $N = 250$, sample from the approximate posterior using our DMM and plot the result in Fig. 2. We compare our method with the semi-automatic ABC (SA-ABC) (Nunes and Prangle, 2015) and Wasserstein SMC (W-SMC) (Bernton et al., 2019) methodologies, as well as Sequential Neural Posterior, Likelihood and Ratio Estimation approaches (SNPE, SNLE and SNRE) (see e.g. Lueckmann et al. (2021)). We see in Fig. 2 that the

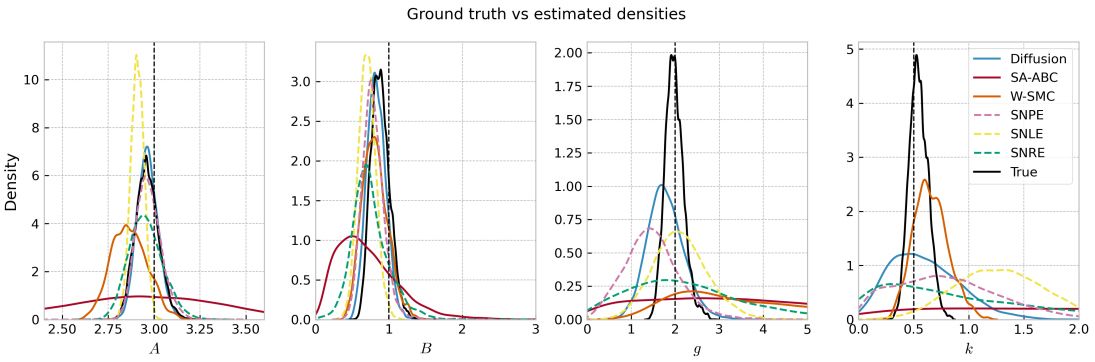

Fig. 2. Posterior kernel density estimates of samples generated using our DMM, SA-ABC, W-SMC, SNLE, SNPE and SNRE for the g-and-k distribution, with $\mathbf{x}_{\text{true}} = (3, 1, 2, 0.5)$ and $N = 250$.

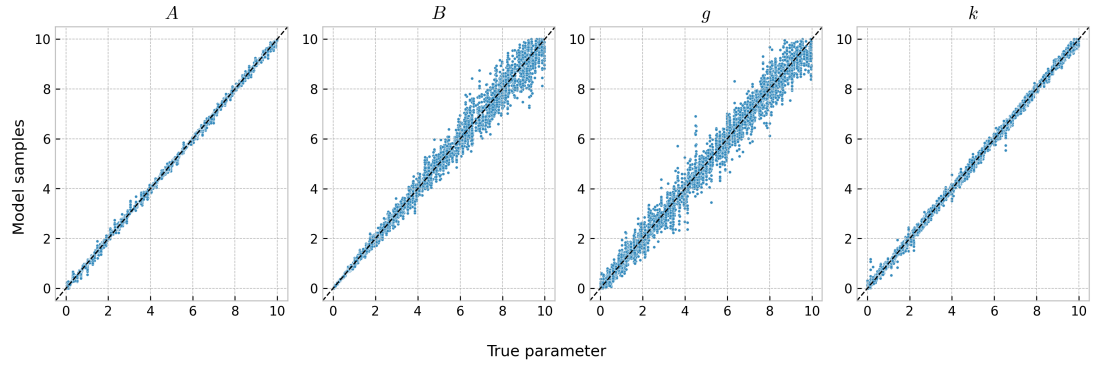

Fig. 3. Comparison of posterior samples $\mathbf{x}_0'$ from our DMM approximation to $p_{\text{data}}(\cdot|\boldsymbol{\xi}_0)$ and the true parameter value $\mathbf{x}_0$ for a range of $\mathbf{x}_0$ in the prior distribution, with $N = 10000$.

DMM achieves more accurate posterior estimation for all parameters, except the kurtosis parameter k for which W-SMC is more accurate. Among the other neural network-based approaches, SNPE appears most competitive on this task, but is less accurate than the DMM especially for parameters g and k. Additional experimental results comparing DMMs to other simulation-based inference methods can be found in (Sharrock et al., 2022; Geffner et al., 2023).

Next, we demonstrate that our model can perform inference for a range of observation values $\boldsymbol{\xi}^*$ simultaneously. We generate a series of 512 parameter values $\mathbf{x}_0$ drawn from $p_{\text{data}}(\mathbf{x}_0)$ and draw an observation $\boldsymbol{\xi}_0$ from $p_{\text{data}}(\boldsymbol{\xi}_0|\mathbf{x}_0)$ with $N = 10000$ for each $\mathbf{x}_0$. Then, we generate 8 samples $\mathbf{x}_0'$ from our approximation to the posterior $p_{\text{data}}(\mathbf{x}_0|\boldsymbol{\xi}_0)$ for each $\boldsymbol{\xi}_0$. We plot each component of the pairs $(\mathbf{x}_0, \mathbf{x}_0')$ in Fig. 3. We see our model is able to infer the original parameters across a range of parameter values.

6.2. Image inpainting and super-resolution using discrete-space CTMCs

Second, we demonstrate that our framework is applicable for large-scale Bayesian inverse problems, such as super-resolution and inpainting for images. For these problems, the prior $p_{\text{data}}(\mathbf{x})$ is the distribution of images. Most ABC techniques such as SA-ABC

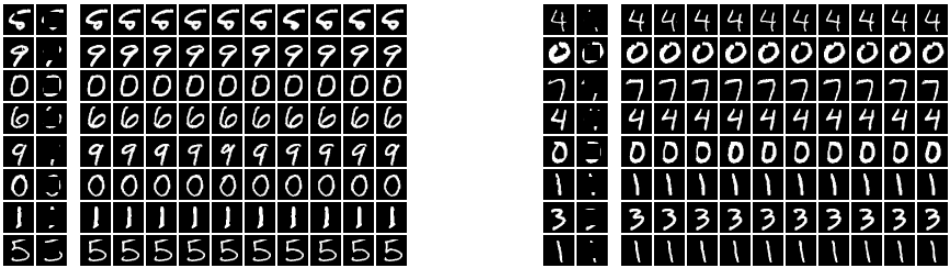

Fig. 4. Samples from the MNIST inpainting task. The first column in each set plots the ground truth images, and the second column has the centre 14×14 pixels missing.

and W-SMC are not applicable as they require an analytical expression for this prior, whereas DMMs do not rely on such an expression.

We consider performing image inpainting for MNIST digit images, where each image $\mathbf{x}_0$ has 28×28 pixels with values in $\{0, \dots, 255\}$, and the observed incomplete image $\boldsymbol{\xi}_0$ has the middle 14×14 pixels missing. Since our state space $\mathcal{X} = \{0, \dots, 255\}^{28 \times 28}$ is discrete, we use the set-up of Example 2 and let the generator of our noising process factor over pixel dimensions. We use the denoising parameterisation of the reverse process (see Appendix F.2) and train by minimising the form of the objective in Example 6.

To test our model, we plot the reconstructed image samples for a number of digits in Fig. 4. We observe that the samples we obtain are consistent with conditioning and appear to be realistic, but also display diversity in the shape of the strokes. In Appendix J.2, we also compare our method to a continuous state space approach.

In addition, we train a conditional discrete-space DMM to perform super-resolution on ImageNet images to demonstrate that this method provides perceptually high quality samples even in very high-dimensional scenarios. For details, see Appendix J.3.

6.3. Modelling distributions on $SO(3)$ using manifold diffusions

Thirdly, we demonstrate that DMMs can approximate distributions on manifolds using two tasks on $SO(3)$. Since $SO(3)$ is a Lie group and so a Riemannian manifold, we use the framework from Example 7. As our noising process, we use Brownian motion with generator $\mathcal{L} = \partial_t + \frac{1}{2}\Delta$. We can explicitly calculate the transition kernels $q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$ for this process, allowing us to use the denoising score matching objective. We parameterise this objective in terms of a neural network approximation $s_\theta(\mathbf{x}, t)$ of the score. This is in contrast to De Bortoli et al. (2022), in which the explicit transition kernels are not used for sampling the forward process or in the loss function, both of which require further approximations.

First we check that our DMM can learn simple mixtures of wrapped normal distributions $p_{\text{data}}(\mathbf{x}) = \frac{1}{M}\sum_{m=1}^{M} \mathcal{N}^W(\mathbf{x}|\mu_m, \sigma_m^2)$, where $\mathcal{N}^W(\mathbf{x} \mid \mu_m, \sigma_m^2)$ is the wrapped normal distribution on $SO(3)$ with expectation μ_m and variance σ_m^2 (De Bortoli et al., 2022). We plot samples from our resulting DMM in Fig. 5. We see that our model provides a good fit to $p_{\text{data}}(\mathbf{x})$, covering all modes. In Appendix J.5, we provide additional results and show that we can also sample from the class conditional density $p_{\text{data}}(\mathbf{x}|m)$.

Second, we consider a more realistic pose estimation task on the SYMSOL dataset, which requires predicting the 3D orientation of various symmetric 3D solids based on

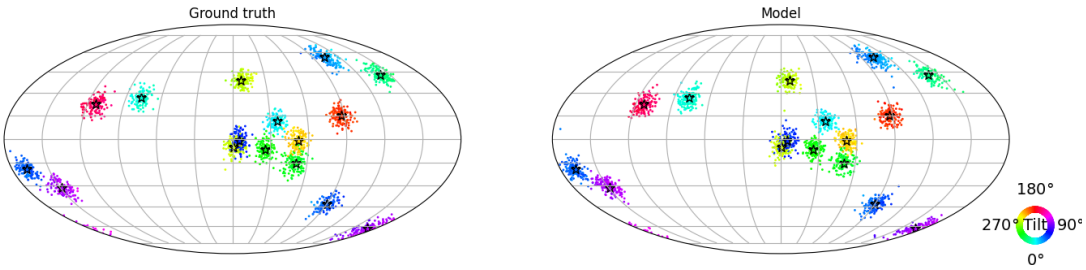

Fig. 5. Samples from the ground truth and our DMM approximation to the mixture of wrapped normal distributions. Each sample is denoted by a point, whose position represents the axis of rotation and whose colour represents the angle of rotation. Stars denote the true cluster means.

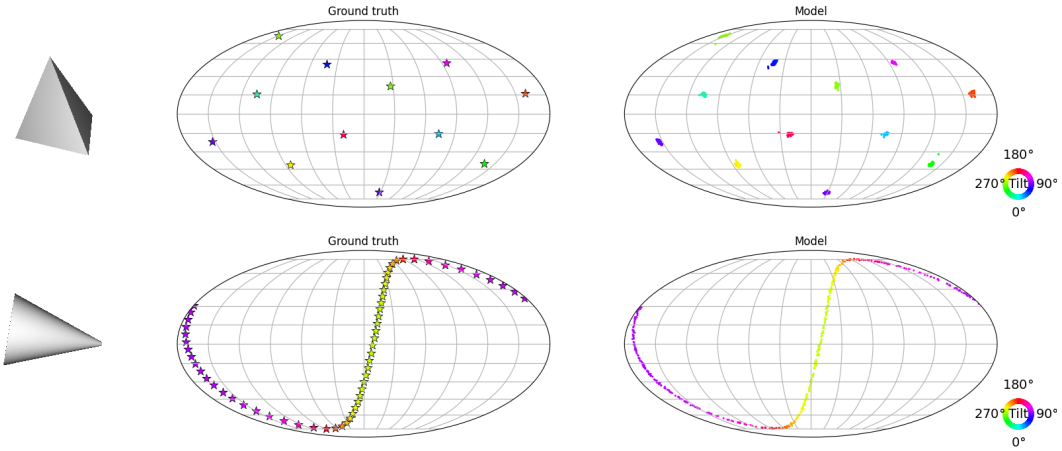

Fig. 6. Samples from the ground truth (plotted as stars, middle) and our pose estimation DMM (right) conditioned on 2D views of two shapes (left). The axis of rotation and rotation angle are represented by position and colour respectively.

2D views (Murphy et al., 2021). Due to the rotational symmetries, a key challenge is to predict all possible poses when only one possibility is presented in training. We use a conditional DMM where $\boldsymbol{\xi}$ is the 2D image view. Fig. 6 shows two sets of samples from our model conditioned on 2D images of two different solids. We see that our model learns to sample from the ground truth accurately and infer the full set of rotational symmetries for different views $\boldsymbol{\xi}$. For further experimental details and plots, see Appendix J.6.

6.4. Approximation of distributions over measures using Wright–Fisher diffusions

Finally, we present an example of learning to approximate a distribution over measures on a finite state space $E = \{1, \dots, N\}$. In this case $\mathcal{X} = \mathcal{P}(E)$, the space of measures on E. This is of particular interest in compositional data analysis (Greenacre, 2021). Elements of $\mathcal{X}$ can be parameterised by tuples of real numbers $\mathbf{p} = (p_1, \dots, p_N) \in [0,1]^N$ such that $\sum_{i=1}^N p_i = 1$. We could approximate the data distribution using a diffusion model on $\mathbb{R}^N$, but such a model would not reflect the fact that our distribution should be supported on a submanifold, the simplex. Using the standard setup for manifold diffusions as in Example 7 would not respect the boundary of the simplex. Other methods have been

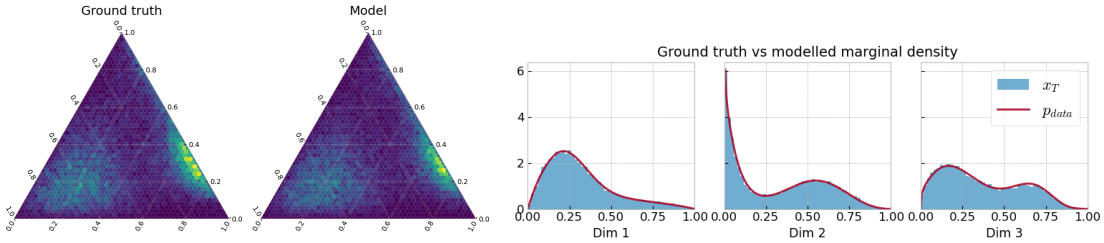

Fig. 7. Histograms of samples from our simplex DMM and the ground truth mixture of Dirichlet distributions for dimension $N = 3$, plotted over the whole space as a ternary plot (left) and over the marginals per dimension (right).

presented in the literature, but they rely on either reflected diffusions (Lou and Ermon, 2023) or on projections of the simplex (Richemond et al., 2022).

We therefore use Wright–Fisher diffusions, a process used in population genetics to model the evolution of allele frequencies, as our class of generative processes. A Wright–Fisher process has generator $\mathcal{L} = \partial_t + \frac{1}{2}\sum_{i,j=1}^{N} p_i\,(\delta_{ij} - p_j)\,\frac{\partial^2}{\partial p_i \partial p_j} + \sum_{i,j=1}^{N} q_{ij} p_i \frac{\partial}{\partial p_j}$, where $(q_{ij})_{i,j=1,\dots,N}$ is some matrix such that $\sum_{j=1}^{N} q_{ij} = 0$ for each $i = 1, \dots, N$. The process takes values in the space of measures on E, and so respects the structure of our data distribution (Ethier and Griffiths, 1993). For specific choices of q_{ij}, the process converges to a known invariant distribution and we can calculate the implicit score matching loss. For details of the theoretical setup, see Appendix F.4.

We evaluate the proposed method by modelling $p_{\text{data}}(\mathbf{x}) = \frac{1}{M}\sum_{m=1}^{M} \text{Dirichlet}(\alpha_m)$, a mixture of Dirichlet distributions with parameters $\alpha_m \in \mathbb{R}^N$, for various values of N. Fig. 7 shows two visualisations of samples drawn from our DMM compared to ground truth samples in dimension $N = 3$. Our model is able to accurately approximate $p_{\text{data}}(\mathbf{x})$. For further evaluations and experimental details, see Appendix J.7.

7. Discussion

We have provided here a general framework which allows us to extend denoising diffusion models to general state-spaces. The resulting DMMs can be trained with principled objectives and used for inference, generalizing along the way score matching ideas. Their applicability and performance have been demonstrated on a range of problems. From a methodological point of view, the proposed framework is general enough to accommodate, for example, general noising processes, mixed continuous/discrete processes and some infinite-dimensional settings with finite representations (though our assumptions on the state space (see Appendix B.1) may fail to hold in the infinite-dimensional setting so more care is required).

However, we still lack a proper theoretical understanding of these models. Under realistic assumptions on the data distribution, De Bortoli (2023) and Chen et al. (2023) show that diffusion models on $\mathbb{R}^d$ can in theory learn essentially any distribution given a good enough score approximation and infinite data. However finite sample guarantees are currently absent. Moreover, p_{data} is typically an empirical measure as we only have access

to a finite set of datapoints, so q_t is a mixture of Gaussians for an Ornstein–Uhlenbeck noising diffusion and its score $\nabla \log q_t$ is thus available. If we were simulating samples using the exact time reversal of this diffusion, we would simply recover the empirical distribution. It is because we are approximating the time-reversal and in particular using an approximation of the scores that we are able to obtain novel samples. It is not yet clear why the approximation of the score using neural networks appears to provide perceptually realistic samples for many applications.

The effectiveness of such methods for inference, even in scenarios where standard MCMC or ABC techniques are not applicable (Sharrock et al., 2022; Geffner et al., 2023), may also be considered surprising. One perspective on the training process is that it involves the model constructing its own summary statistics that allow it to perform inference effectively on the training observations. It is not yet well understood why the summary statistics the model learns appear empirically effective, or what sorts of summary statistics our training procedure biases the model towards.

Overall, this contribution shows how the range of existing models relate to each other and may help applying DMMs in practice to a large variety of problems. However, our understanding of such models is still incomplete and deserves further attention.

Acknowledgments

Joe Benton was supported by the EPSRC Centre for Doctoral Training in Modern Statistics and Statistical Machine Learning (EP/S023151/1) and Yuyang Shi by the Huawei UK Fellowship Programme. Arnaud Doucet acknowledges support of the UK Dstl and EPSRC grant EP/R013616/1. This is part of the collaboration between US DOD, UK MOD and UK EPSRC under the Multidisciplinary University Research Initiative. He also acknowledges support from the EPSRC grants CoSines (EP/R034710/1) and Bayes4Health (EP/R018561/1).

References

Anderson, B. D. O. (1982). Reverse-time Diffusion Equation Models. *Stochastic Processes and their Applications 12*, 313–326.

Austin, J., D. D. Johnson, J. Ho, D. Tarlow, and R. van den Berg (2021). Structured Denoising Diffusion Models in Discrete State-Spaces. *NeurIPS*.

Bernton, E., P. E. Jacob, M. Gerber, and C. P. Robert (2019). Approximate Bayesian Computation with the Wasserstein Distance. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 81*(2), 235–269.

Brown, T., B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. (2020). Language Models are Few-shot Learners. *NeurIPS*.

Campbell, A., J. Benton, V. De Bortoli, T. Rainforth, G. Deligiannidis, and A. Doucet (2022). A Continuous Time Framework for Discrete Denoising Models. *NeurIPS*.

Chen, S., S. Chewi, J. Li, Y. Li, A. Salim, and A. R. Zhang (2023). Sampling is as Easy as Learning the Score: Theory for Diffusion Models with Minimal Data Assumptions. *ICLR*.

De Bortoli, V. (2023). Convergence of Denoising Diffusion Models under the Manifold Hypothesis. *Transactions on Machine Learning Research*.

De Bortoli, V., E. Mathieu, M. Hutchinson, J. Thornton, Y. W. Teh, and A. Doucet (2022). Riemannian Score-Based Generative Modeling. *NeurIPS*.

Dhariwal, P. and A. Nichol (2021). Diffusion Models Beat GANs on Image Synthesis. *NeurIPS*.

Ethier, S. N. and R. C. Griffiths (1993). The Transition Function of a Fleming-Viot Process. *The Annals of Probability 21*, 1571–1590.

Geffner, T., G. Papamakarios, and A. Mnih (2023). Compositional Score Modeling for Simulation-based Inference. *arXiv preprint arXiv:2209.14249*.

Goodfellow, I. J., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio (2014). Generative Adversarial Nets. *NeurIPS*.

Greenacre, M. (2021). Compositional Data Analysis. *Annual Review of Statistics and its Application 8*, 271–299.

Gutmann, M. U. and J.-i. Hirayama (2011). Bregman Divergence as General Framework to Estimate Unnormalized Statistical Models. *UAI*.

Ho, J., A. Jain, and P. Abbeel (2020). Denoising Diffusion Probabilistic Models. *NeurIPS*.

Hoogeboom, E., D. Nielsen, P. Jaini, P. Forré, and M. Welling (2021). Argmax Flows and Multinomial Diffusion: Learning Categorical Distributions. *NeurIPS*.

Huang, C.-W., M. Aghajohari, A. J. Bose, P. Panangaden, and A. Courville (2022). Riemannian Diffusion Models. *NeurIPS*.

Huang, C.-W., J. H. Lim, and A. Courville (2021). A Variational Perspective on Diffusion-Based Generative Models and Score Matching. *NeurIPS*.

Hyvärinen, A. (2005). Estimation of Non-Normalized Statistical Models by Score Matching. *Journal of Machine Learning Research 6*, 695–709.

Hyvärinen, A. (2007). Some Extensions of Score Matching. *Computational Statistics and Data Analysis 51*, 2499 – 2512.

Kingma, D. P. and M. Welling (2014). Auto-Encoding Variational Bayes. *ICLR*.

Lou, A. and S. Ermon (2023). Reflected Diffusion Models. *arXiv preprint arXiv:2304.04740*.

Lueckmann, J.-M., J. Boelts, D. S. Greenberg, P. J. Gonçalves, and J. H. Macke (2021). Benchmarking Simulation-Based Inference. *AISTATS*.

Lyu, S. (2009). Interpretation and Generalization of Score Matching. *UAI*.

Mardia, K. V., J. T. Kent, and A. K. Laha (2016). Score Matching Estimators for Directional Distributions. *arXiv preprint arXiv:1604.08470*.

Murphy, K. A., C. Esteves, V. Jampani, S. Ramalingam, and A. Makadia (2021). Implicit-PDF: Non-Parametric Representation of Probability Distributions on the Rotation Manifold. *ICML*, 7882–7893.

Nunes, M. A. and D. Prangle (2015). abctools: An R Package for Tuning Approximate Bayesian Computation Analyses. *The R Journal 7*(2), 189–205.

Oord, A. v. d., S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu (2016). WaveNet: A Generative Model for Raw Audio. *arXiv:1609.03499*.

Popov, V., I. Vovk, V. Gogoryan, T. Sadekova, and M. Kudinov (2021). Grad-tts: A Diffusion Probabilistic Model for Text-to-speech. *ICML*.

Rezende, D. J. and S. Mohamed (2015). Variational Inference with Normalizing Flows. *ICML*.

Richemond, P. H., S. Dieleman, and A. Doucet (2022). Categorical SDEs with Simplex Diffusion. *arXiv preprint arXiv:2210.14784*.

Saharia, C., J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi (2022). Image Super-Resolution via Iterative Refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–14.

Sharrock, L., J. Simons, S. Liu, and M. Beaumont (2022). Sequential Neural Score Estimation: Likelihood-Free Inference with Conditional Score Based Diffusion Models. *arXiv preprint arXiv:2210.04872*.

Sohl-Dickstein, J., E. A. Weiss, N. Maheswaranathan, and S. Ganguli (2015). Deep Unsupervised Learning Using Nonequilibrium Thermodynamics. *ICML*.

Song, Y., C. Durkan, I. Murray, and S. Ermon (2021). Maximum Likelihood Training of Score-Based Diffusion Models. *NeurIPS*.

Song, Y., J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole (2021). Score-Based Generative Modeling through Stochastic Differential Equations. *ICLR*.

Sun, H., L. Yu, B. Dai, D. Schuurmans, and H. Dai (2023). Score-based Continuous-time Discrete Diffusion Models. *ICLR*.

Trippe, B. L., J. Yim, D. Tischer, D. Baker, T. Broderick, R. Barzilay, and T. Jaakkola (2023). Diffusion Probabilistic Modeling of Protein Backbones in 3D for the Motif-scaffolding Problem. *ICLR*.

Vincent, P. (2011). A Connection Between Score Matching and Denoising Autoencoders. *Neural Computation 23*, 1661–1674.

Yu, S., M. Drton, and A. Shojaie (2022). Generalized Score Matching for General Domains. *Information and Inference: A Journal of the IMA 11*(2), 739–780.

A. Background on Feller processes

We recall some basic definitions and properties associated with Feller processes which we use for the derivations in Section 3. Our principal source is Dong (2003).

A.1. Definition of a Feller process

Let S be a locally compact, separable metric space and let $C_0(S)$ denote the set of continuous functions $f : S \to \mathbb{R}$ such that for any $\epsilon > 0$ there exists a compact $K \subseteq S$ such that $|f(x)| < \epsilon$ for all $x \notin K$. Also, let $||f||$ denote the supremum norm on $C_0(S)$.

Definition 1 (Feller process). *A time-homogeneous Markov process $(X_t)_{t\geq 0}$ with state space S and associated transition semigroup $(P_t)_{t\geq 0}$ is a Feller process if:*

- *$P_t f \in C_0(S)$ for all $f \in C_0(S)$ and $t \geq 0$.*
- *$||P_t f|| \leq ||f||$ for all $f \in C_0(S)$.*
- *$P_t f(x) \to f(x)$ as $t \to 0$ for all $x \in S$ and $f \in C_0(S)$.*

Definition 2 (Generator of a Feller process). *Suppose X is a Feller process on S as above and f is a function in $C_0(S)$. If the limit*

$$\mathcal{A}f := \lim_{s\to 0} \frac{P_s f - f}{s}$$

exists in $C_0(S)$, we say that f is in the domain of the generator of X. We call the operator $\mathcal{A}$ defined in this way the generator of X and denote its domain by $\mathcal{D}(\mathcal{A})$.

In the main text, we are concerned with Feller processes $\overline{X}, \overline{Y}$ defined on the extended space $\mathcal{S} = \mathcal{X} \times [0, \infty)$ which are constructed by taking a time-inhomogeneous Markov process X on $\mathcal{X}$ and defining $\overline{X} = (X_t, t)_{t\geq 0}$. In this setting, we have the following variant of Dynkin's formula.

Lemma 3 (Dynkin's formula). *If $\overline{X} = (X_t, t)_{t\geq 0}$ is a Feller process on $\mathcal{S}$ with generator $\mathcal{A}$ and $f \in \mathcal{D}(\mathcal{A})$, then*

$$M_t^f = f(X_t, t) - f(X_0, 0) - \int_0^t \mathcal{A}f(X_s, s)\ \mathrm{d}s$$

is a martingale with respect to the natural filtration of $\overline{X}$.

Proof. See Theorem 27.20 in Dong (2003).

A.2. Adjoint of a generator

Given a state space S and a reference measure ν on S, we can define an inner product on $C_0(S)$ by letting

$$\langle f, h\rangle = \int_S fh\ \mathrm{d}\nu$$

for all $f, h \in C_0(S)$ such that the integral exists. This induces a Hilbert space structure on $C_0(S)$ and allows us to make the following definition, from Yosida (1965).

DEFINITION 3 (ADJOINT OF AN OPERATOR). *Given operator $\mathcal{A}$ with domain $\mathcal{D}(\mathcal{A})$ contained in $C_0(S)$, we define the adjoint operator $\mathcal{A}^*$ acting at function $f \in C_0(S)$ by*

$$\langle \mathcal{A}^* f, h \rangle = \langle f, \mathcal{A} h \rangle \qquad \text{for all } h \in \mathcal{D}(\mathcal{A}).$$

The domain $\mathcal{D}(\mathcal{A}^*)$ of $\mathcal{A}^*$ is the set of all functions f such that there exists some function $\mathcal{A}^* f$ for which the above holds.

B. Assumptions for Section 3

Here, we list the assumptions under which our derivations in Section 3 hold. Note that these assumptions can be verified in several relevant cases (see Appendix F).

B.1. Assumptions on the state space $\mathcal{X}$

ASSUMPTION 3. *Our state space $\mathcal{X}$ is a locally compact, separable metric space. In addition, there exists a reference measure ν on $\mathcal{X}$ with respect to which all relevant probability distributions are absolutely continuous.*

B.2. Assumptions on the marginals p and q

ASSUMPTION 4. *We have $p_t \in \mathcal{D}(\hat{\mathcal{K}}^*)$ for each $t \in [0, T]$, where $\hat{\mathcal{K}}^*$ is the adjoint of the spatial part of the operator $\mathcal{K}$. In addition, p is differentiable with respect to t and $\partial_t p$ is bounded.*

B.3. Assumptions on the generators $\mathcal{K}$ and $\mathcal{L}$

ASSUMPTION 5. *$\overline{X}$ and $\overline{Y}$ are Feller processes with associated transition semigroups $(P_t)_{t \geq 0}$, $(Q_t)_{t \geq 0}$ and generators $\mathcal{K}, \mathcal{L}$ respectively.*

ASSUMPTION 6. *$\mathcal{K}$ decomposes as $\mathcal{K} = \partial_t + \hat{\mathcal{K}}$, where $\hat{\mathcal{K}} f$ is defined only in terms of the spatial arguments of f, so we may view it as an operator on (a subset of) $C_0(\mathcal{X})$.*

ASSUMPTION 7. *There exists a subset $\mathcal{D}_0 \subseteq \mathcal{D}(\hat{\mathcal{K}}) \cap L^2(\mathcal{X}, \nu)$ which is dense in $L^2(\mathcal{X}, \nu)$, satisfies $\hat{\mathcal{K}} h \in \mathcal{D}_0$ for all $h \in \mathcal{D}_0$ and such that every function in $\mathcal{D}_0$ is bounded and has compact support.*

B.4. Assumptions on $\mathcal{M}$ and c

ASSUMPTION 8. *The function $c : \mathcal{S} \to \mathbb{R}$ is bounded, and the function $v : \mathcal{S} \to \mathbb{R}$ is bounded, in $\mathcal{D}(\mathcal{M})$ and satisfies $\int_0^T \mathbb{E}\left[|\mathcal{M} v(Z_s, s)|^2\right] \mathrm{d}s < \infty$.*

B.5. Assumptions on β

ASSUMPTION 9. *The functions β^{-1}, $\beta^{-1} v$, $\log \beta$ and $\log v$ are in $\mathcal{D}(\mathcal{L})$, β^{-1} and $\beta \mathcal{L}(\beta^{-1})$ are both bounded, and $\beta \in \mathcal{D}(\hat{\mathcal{L}}^*)$.*

C. Stochastic process theory

We provide full statements of the general stochastic process results used in Section 3. For completeness, we also provide proofs of the given results adapted to our setting.

THEOREM 4 (FOKKER–PLANCK). *Let $(X_t)_{t\in[0,T]}$ be a Markov process with generator $\mathcal{K}$ and marginals p_t satisfying the assumptions in Appendix B. Then p satisfies the forward Kolmogorov equation $\partial_t p = \hat{\mathcal{K}}^* p$ for ν-almost every $\mathbf{x}$.*

PROOF. For any $h \in \mathcal{D}_0$, by Assumptions 4 and 7 we may write

$$\begin{aligned}\langle \partial_t p - \hat{\mathcal{K}}^* p, h\rangle &= \int_{\mathcal{X}} (\partial_t p)h - p(\hat{\mathcal{K}}h)\,\mathrm{d}\nu \\ &= \partial_t \mathbb{E}\left[h(X_t)\right] - \mathbb{E}\left[\hat{\mathcal{K}}h(X_t)\right].\end{aligned}$$

Applying Dynkin's formula to $f(\mathbf{x}, t) = h(\mathbf{x})$, taking expectations and using Fubini's theorem, we see that

$$\mathbb{E}\left[h(X_t)\right] - \mathbb{E}\left[h(X_0)\right] = \int_0^t \mathbb{E}\left[\hat{\mathcal{K}}h(X_s)\right]\,\mathrm{d}s.$$

Differentiating with respect to t, we deduce that $\langle \partial_t p - \hat{\mathcal{K}}^* p, h\rangle = 0$. Since this holds for all $h \in \mathcal{D}_0$ and $\mathcal{D}_0$ is dense in $L^2(\mathcal{X}, \nu)$, we conclude that $\partial_t p - \hat{\mathcal{K}}^* p = 0$ holds ν-a.e. as required.

THEOREM 5 (FEYNMAN–KAC). *Let $\overline{Z} = (Z_t, t)_{t\geq 0}$ be a Feller process on $\mathcal{S}$ with generator $\mathcal{M}$. Suppose that we are given functions $v, c : \mathcal{S} \to \mathbb{R}$ and $h : \mathcal{X} \to \mathbb{R}$ such that $\mathcal{M}, v, c$ solve equation $\mathcal{M}v + cv = 0$ (as in Assumption 1) with boundary condition $v(\cdot, T) = h(\cdot)$. Suppose also that Assumption 8 is satisfied. Then we have*

$$v(\mathbf{x}, \tau) = \mathbb{E}\left[h(Z_T)\exp\left\{\int_\tau^T c(Z_s, s)\,\mathrm{d}s\right\} \;\middle|\; Z_\tau = \mathbf{x}\right]$$

for all $0 \leq \tau \leq T$.

PROOF. This result is well-known in the case of Euclidean diffusion processes (Karatzas and Shreve, 1991). In the general case, the proof relies on the theory of semimartingales (see for example Métivier (1982)). Fix $\tau \in [0, T]$ and for all $t \in [\tau, T]$ define

$$S_t = v(Z_t, t)\exp\left\{\int_\tau^t c(Z_s, s)\mathrm{d}s\right\}$$

along with

$$V_t = v(Z_t, t), \qquad U_t = \exp\left\{\int_\tau^t c(Z_s, s)\mathrm{d}s\right\}.$$

Each of these processes is clearly a semimartingale, and so we may define $\mathrm{d}S_t$, $\mathrm{d}U_t$ and $\mathrm{d}V_t$ accordingly (Métivier, 1982). The following lemma will allow us to express $\mathrm{d}S_t$ in terms of $\mathrm{d}U_t$ and $\mathrm{d}V_t$.

LEMMA 4 (INTEGRATION BY PARTS FOR SEMIMARTINGALES). *If U and V are semimartingales and at least one is continuous then we have*

$$\mathrm{d}(U_tV_t) = U_{t-}\mathrm{d}V_t + U_{t-}\mathrm{d}V_t + \mathrm{d}[U,V]^c_t,$$

where $[\cdot,\cdot]^c_t$ denotes the quadratic covariation.

PROOF. This is Theorem 2.7.4(ii) of Pulido (2011), or follows from applying Theorem 27.1 of Métivier (1982) to the function $\varphi(U,V) = UV$.

Since $v \in \mathcal{D}(\mathcal{M})$ by Assumption 8, by Dynkin's formula we have that V is a semimartingale and we may decompose

$$\mathrm{d}V_t = \mathcal{M}v\mathrm{d}t + \mathrm{d}M^v_t$$

where M^v_t is a martingale. Also, since $c(x,t)$ is bounded by Assumption 8, U is a continuous, adapted, previsible process of finite variation and satisfies

$$\mathrm{d}U_t = c(Z_t,t)\exp\left\{\int_\tau^t c(Z_s,s)\,\mathrm{d}s\right\}\mathrm{d}t.$$

In addition, note that $\mathrm{d}[U,V]^c_t = 0$ since U is continuous and of finite variation. Therefore, by Lemma 4, we can calculate

$$\begin{aligned} S_t - S_\tau &= \int_\tau^t U_{s-}\mathrm{d}V_s + \int_\tau^t V_{s-}\mathrm{d}U_s + [U,V]^c_t \\ &= \int_\tau^t U_s\big\{\mathcal{M}v + cv\big\}\mathrm{d}s + \int_\tau^t U_s\,\mathrm{d}M^v_s \\ &= \int_\tau^t U_s\,\mathrm{d}M^v_s \end{aligned}$$

where we have used that $\mathcal{M}v + cv = 0$ in the last line. Therefore, S can be expressed as a stochastic integral with respect to the martingale M^v.

The conditions we have imposed through Assumption 8 on c and v imply that U is bounded and M^v is square-integrable. It follows, for example from Theorem 24.4.5 in (Métivier, 1982), that S is a local martingale and hence, since it is also bounded, a true martingale. We then have that

$$\begin{aligned} v(\mathbf{x},\tau) &= \mathbb{E}\left[S_\tau | Z_\tau = \mathbf{x}\right] = \mathbb{E}\left[S_T | Z_\tau = \mathbf{x}\right] \\ &= \mathbb{E}\left[h(Z_T)\exp\left\{\int_\tau^T c(Z_s,s)\mathrm{d}s\right\}\middle| Z_\tau = \mathbf{x}\right] \end{aligned}$$

as required.

THEOREM 6 (GIRSANOV). *Let $\overline{Y} = (Y_t,t)_{t\geq 0}$ and $\overline{Z} = (Z_t,t)_{t\geq 0}$ be Feller processes on $\mathcal{S}$ with generators $\mathcal{L}$, $\mathcal{M}$ and path measures $\mathbb{Q}$, $\mathbb{P}$ respectively, such that Y_0 and Z_0 have the same law. Suppose also that there exists a bounded, measurable function $\alpha : \mathcal{S} \to (0,\infty)$ in $\mathcal{D}(\mathcal{L})$ such that $\alpha^{-1}\mathcal{L}\alpha$ is bounded, and such that*

$$\alpha\mathcal{M}f = \mathcal{L}(f\alpha) - f\mathcal{L}\alpha \tag{12}$$

for all functions f such that $f \in \mathcal{D}(\mathcal{M})$ and $f\alpha \in \mathcal{D}(\mathcal{L})$. Then we have

$$\frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\mathbb{Q}}(\omega) = \frac{\alpha(\omega_T, t)}{\alpha(\omega_0, 0)} \exp\Big\{ - \int_0^T \frac{\mathcal{L}\alpha(\omega_s, s)}{\alpha(\omega_s, s)} \, \mathrm{d}s \Big\}. \tag{13}$$

Proof. This essentially follows from the work of Palmowski and Rolski (2002). Using their terminology, their Proposition 3.2 implies α is a good function, so the RHS of Equation (13) is a martingale and we may define a measure $\tilde{\mathbb{P}}$ by

$$\frac{\mathrm{d}\tilde{\mathbb{P}}}{\mathrm{d}\mathbb{Q}}(\omega) = \frac{\alpha(\omega_T, t)}{\alpha(\omega_0, 0)} \exp\Big\{ - \int_0^T \frac{\mathcal{L}\alpha(\omega_s, s)}{\alpha(\omega_s, s)} \, \mathrm{d}s \Big\}.$$

Under the measure $\tilde{\mathbb{P}}$, the canonical process $(\omega_t)_{t\in[0,T]}$ is still Markov. By the proof of their Theorem 4.2, we see that

$$\tilde{D}_t^f = f(Y_t, t) - \int_0^t \mathcal{M}f(Y_s, s)\mathrm{d}s$$

is a martingale for all sufficiently smooth functions f, implying that $\mathcal{M}$ is the generator of $(\omega_t)_{t\in[0,T]}$ under $\tilde{\mathbb{P}}$. It follows that $(\omega_t)_{t\in[0,T]}$ has the same law under $\tilde{\mathbb{P}}$ as $\overline{Z}$ does under $\mathbb{Q}$, which is sufficient to prove the result since $\overline{Y}$ and $\overline{Z}$ are Feller.

D. Proof from Section 3

We give the proofs of Lemma 1 and Theorem 2 from Section 3.

Lemma 1. *Let the generator $\mathcal{L}$ and the functions β and c be as above. Then, we have* $v^{-1}\beta\mathcal{L}(\beta^{-1}v) + \mathcal{L}\log\beta = \beta^{-1}\hat{\mathcal{L}}^*\beta + \hat{\mathcal{L}}\log\beta$.

Proof. Let us define $\hat{\mathcal{M}}$ to be the operator such that $\mathcal{M} = \hat{\mathcal{M}} + \partial_t$. Then, since $\hat{\mathcal{M}} + c = \mathcal{M} + c - \partial_t = \hat{\mathcal{K}}^*$, for any sufficiently rapidly decaying test function f we have

$$\langle \hat{\mathcal{M}}f, 1\rangle + \langle cf, 1\rangle = \langle \hat{\mathcal{K}}^* f, 1\rangle = \langle f, \hat{\mathcal{K}}1\rangle = 0,$$

so $\langle \hat{\mathcal{M}}f, 1\rangle = -\langle c, f\rangle$. Assumption 2, which states that $\beta^{-1}\mathcal{M}f = \mathcal{L}(\beta^{-1}f) - f\mathcal{L}(\beta^{-1})$ for all sufficiently rapidly decaying f, can be rearranged to $\hat{\mathcal{M}}f = \beta\hat{\mathcal{L}}(\beta^{-1}f) - \beta f\hat{\mathcal{L}}(\beta^{-1})$. So, it follows that

$$\begin{aligned}\langle c, f\rangle &= -\langle \beta\hat{\mathcal{L}}(\beta^{-1}f), 1\rangle + \langle \beta f\hat{\mathcal{L}}(\beta^{-1}), 1\rangle \\ &= -\langle f, \beta^{-1}\hat{\mathcal{L}}^*\beta\rangle + \langle f, \beta\hat{\mathcal{L}}(\beta^{-1})\rangle\end{aligned}$$

for any sufficiently rapidly decaying f. We conclude that $\beta^{-1}\hat{\mathcal{L}}^*\beta = \beta\hat{\mathcal{L}}(\beta^{-1}) - c$.

Next, using Assumption 2 with $f = v$ we can write

$$\begin{aligned}v^{-1}\beta\mathcal{L}(\beta^{-1}v) &= \beta\mathcal{L}(\beta^{-1}) + v^{-1}\mathcal{M}v \\ &= \beta\mathcal{L}(\beta^{-1}) - c \\ &= -\partial_t\log\beta + \beta\hat{\mathcal{L}}(\beta^{-1}) - c \\ &= -\partial_t\log\beta + \beta^{-1}\hat{\mathcal{L}}^*\beta.\end{aligned}$$

Finally, note that $-\mathcal{L}\log\beta + \hat{\mathcal{L}}\log\beta = -\partial_t \log\beta$. Combining this with the final line above, we get the desired result.

THEOREM 2. *With the above set-up, minimising the objective*

$$\mathcal{I}_{\mathrm{DSM}}(\beta) = \int_0^T \mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t,\boldsymbol{\xi}_0)}\left[\frac{\mathcal{L}(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\boldsymbol{\xi}_0,\cdot))(\mathbf{x}_t,t)}{q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)/\beta(\mathbf{x}_t,\boldsymbol{\xi}_0,t)} - \mathcal{L}\log(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\boldsymbol{\xi}_0,\cdot))(\mathbf{x}_t,t)\right]\mathrm{d}t$$

is equivalent to maximising a lower bound on the expected model log-likelihood.

PROOF. Applying Theorem 1 to the generative process $X^{\boldsymbol{\xi}^*}$ conditioned on observation $\boldsymbol{\xi}^*$,

$$\log p_T(\mathbf{x}_0|\boldsymbol{\xi}^*) \geq \mathbb{E}_{\mathbb{Q}}\Big[\log p_0(Y_T)\Big|Y_0 = \mathbf{x}_0\Big] - \int_0^T \mathbb{E}_{\mathbb{Q}}\left[\frac{\hat{\mathcal{L}}^*\beta(\mathbf{x}_t,\boldsymbol{\xi}^*,t)}{\beta(\mathbf{x}_t,\boldsymbol{\xi}^*,t)} + \hat{\mathcal{L}}\log\beta(\mathbf{x}_t,\boldsymbol{\xi}^*,t)\;\middle|\; Y_0 = \mathbf{x}_0\right]\mathrm{d}s.$$

Replacing $\boldsymbol{\xi}^*$ by $\boldsymbol{\xi}_0$, letting $(\mathbf{x}_0,\boldsymbol{\xi}_0) \sim p_{\mathrm{data}}$ and taking expectations, we get

$$\mathbb{E}_{p_{\mathrm{data}}(\mathbf{x}_0,\boldsymbol{\xi}_0)}\left[\log p_T(\mathbf{x}_0|\boldsymbol{\xi}_0)\right] \geq \mathbb{E}_{q_T(\mathbf{x}_T)}\Big[\log p_0(\mathbf{x}_T)\Big] - \int_0^T \mathbb{E}_{q(\mathbf{x}_t,\boldsymbol{\xi}_0)}\left[\frac{\hat{\mathcal{L}}^*\beta(\mathbf{x}_t,\boldsymbol{\xi}_0,t)}{\beta(\mathbf{x}_t,\boldsymbol{\xi}_0,t)} + \hat{\mathcal{L}}\log\beta(\mathbf{x}_t,\boldsymbol{\xi}_0,t)\right]\mathrm{d}s.$$

For any given $\boldsymbol{\xi}$, we have

$$\begin{aligned}&\mathbb{E}_{q(\mathbf{x}_t|\boldsymbol{\xi})}\left[\frac{\hat{\mathcal{L}}^*\beta(\mathbf{x}_t,\boldsymbol{\xi},t)}{\beta(\mathbf{x}_t,\boldsymbol{\xi},t)} + \hat{\mathcal{L}}\log\beta(\mathbf{x}_t,\boldsymbol{\xi},t)\right]\\ &= \mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t|\boldsymbol{\xi})}\left[\frac{\mathcal{L}(q_{\cdot|0}(\cdot|\mathbf{x}_0,\boldsymbol{\xi})/\beta(\cdot,\boldsymbol{\xi},\cdot))(\mathbf{x}_t,t)}{q_{t|0}(\mathbf{x}_t|\mathbf{x}_0,\boldsymbol{\xi})/\beta(\mathbf{x}_t,\boldsymbol{\xi},t)} - \mathcal{L}\log(q_{\cdot|0}(\cdot|\mathbf{x}_0,\boldsymbol{\xi})/\beta(\cdot,\boldsymbol{\xi},\cdot))(\mathbf{x}_t,t)\right] + const\end{aligned}$$

by the argument of Appendix E (see below), where the constant depends only on the dynamics of the forward process. Substituting $\boldsymbol{\xi}_0$ for $\boldsymbol{\xi}$ and taking expectations over $\boldsymbol{\xi}_0 \sim p_{\mathrm{data}}$, noting that $q_{t|0}(\mathbf{x}_t|\mathbf{x}_0,\boldsymbol{\xi}_0) = q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$, we get

$$\begin{aligned}&\mathbb{E}_{q(\mathbf{x}_t,\boldsymbol{\xi}_0)}\left[\frac{\hat{\mathcal{L}}^*\beta(\mathbf{x}_t,\boldsymbol{\xi}_0,t)}{\beta(\mathbf{x}_t,\boldsymbol{\xi}_0,t)} + \hat{\mathcal{L}}\log\beta(\mathbf{x}_t,\boldsymbol{\xi}_0,t)\right]\\ &= \mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t,\boldsymbol{\xi}_0)}\left[\frac{\mathcal{L}(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\boldsymbol{\xi}_0,\cdot))(\mathbf{x}_t,t)}{q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)/\beta(\mathbf{x}_t,\boldsymbol{\xi}_0,t)} - \mathcal{L}\log(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\boldsymbol{\xi}_0,\cdot))(\mathbf{x}_t,t)\right] + const.\end{aligned}$$

It follows that

$$\begin{aligned}&\mathbb{E}_{p_{\mathrm{data}}(\mathbf{x}_0,\boldsymbol{\xi}_0)}\left[\log p_T(\mathbf{x}_0|\boldsymbol{\xi}_0)\right] \geq \mathbb{E}_{q_T(\mathbf{x}_T)}\Big[\log p_0(\mathbf{x}_T)\Big]\\ &- \int_0^T \mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t,\boldsymbol{\xi}_0)}\left[\frac{\mathcal{L}(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\boldsymbol{\xi}_0,\cdot))(\mathbf{x}_t,t)}{q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)/\beta(\mathbf{x}_t,\boldsymbol{\xi}_0,t)} - \mathcal{L}\log(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\boldsymbol{\xi}_0,\cdot))(\mathbf{x}_t,t)\right]\mathrm{d}s\\ &\qquad + const.\end{aligned}$$

The first term on the RHS and the constant are independent of the dynamics of the reverse process. Hence minimising

$$\mathcal{I}_{\mathrm{DSM}}(\beta) = \int_0^T \mathbb{E}_{q(\mathbf{x}_0,\mathbf{x}_t,\boldsymbol{\xi}_0)} \left[\frac{\mathcal{L}(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\boldsymbol{\xi}_0,\cdot))(\mathbf{x}_t,t)}{q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)/\beta(\mathbf{x}_t,\boldsymbol{\xi}_0,t)} - \mathcal{L}\log(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\boldsymbol{\xi}_0,\cdot))(\mathbf{x}_t,t) \right] \mathrm{d}t$$

is equivalent to maximising a lower bound on $\mathbb{E}_{p_{\mathrm{data}}(\mathbf{x}_0,\boldsymbol{\xi}_0)}\left[\log p_T(\mathbf{x}_0|\boldsymbol{\xi}_0)\right]$, which is the expected model log-likelihood.

E. Equivalence of generalised score matching objectives

First, we show that $\mathcal{I}_{\mathrm{ISM}}$ and $\mathcal{I}_{\mathrm{DSM}}$ are equivalent training objectives.

$$\begin{aligned}
&\mathbb{E}_{q_{0,t}(\mathbf{x}_0,\mathbf{x}_t)} \left[\frac{\mathcal{L}(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\cdot))(\mathbf{x}_t,t)}{q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)/\beta(\mathbf{x}_t,t)} - \mathcal{L}\log(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\cdot))(\mathbf{x}_t,t) \right] \\
&= \int_{\mathcal{X}} \int_{\mathcal{X}} q_{0,t}(\mathbf{x}_0,\mathbf{x}_t) \left\{ \frac{\mathcal{L}(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\cdot))(\mathbf{x}_t,t)}{q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)/\beta(\mathbf{x}_t,t)} - \mathcal{L}\log(q_{\cdot|0}(\cdot|\mathbf{x}_0)/\beta(\cdot,\cdot))(\mathbf{x}_t,t) \right\} \mathrm{d}\nu(\mathbf{x}_0)\mathrm{d}\nu(\mathbf{x}_t) \\
&= \int_{\mathcal{X}} q_0(\mathbf{x}_0) \int_{\mathcal{X}} \left\{ \beta(\mathbf{x}_t,t)\hat{\mathcal{L}} \left(\frac{q_{t|0}(\cdot|\mathbf{x}_0)}{\beta(\cdot,t)} \right)(\mathbf{x}_t) - q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)\hat{\mathcal{L}}\log \left(\frac{q_{t|0}(\cdot|\mathbf{x}_0)}{\beta(\cdot,t)} \right)(\mathbf{x}_t) \right\} \mathrm{d}\nu(\mathbf{x}_t)\mathrm{d}\nu(\mathbf{x}_0) \\
&= \int_{\mathcal{X}} q_0(\mathbf{x}_0) \int_{\mathcal{X}} q_{t|0}(\mathbf{x}_t|\mathbf{x}_0) \left\{ \frac{\hat{\mathcal{L}}^*\beta(\mathbf{x}_t,t)}{\beta(\mathbf{x}_t,t)} + \hat{\mathcal{L}}\log\beta(\mathbf{x}_t,t) \right\} \mathrm{d}\nu(\mathbf{x}_t)\mathrm{d}\nu(\mathbf{x}_0) + const \\
&= \mathbb{E}_{q_t(\mathbf{x}_t)} \left[\frac{\hat{\mathcal{L}}^*\beta}{\beta} + \hat{\mathcal{L}}\log\beta \right] + const,
\end{aligned}$$

where the constants depend only on the dynamics of the forward process and so are fixed during training. Integrating from $t = 0$ to $t = T$, we conclude that $\mathcal{I}_{\mathrm{ISM}}$ and $\mathcal{I}_{\mathrm{DSM}}$ are equivalent.

There is also an *explicit score matching* form of the general DMM training objective as follows:

$$\mathcal{I}_{\mathrm{ESM}}(\beta) = \int_0^T \mathbb{E}_{q_t(\mathbf{x}_t)} \left[\frac{\mathcal{L}(q_{\cdot}(\cdot)/\beta(\cdot,\cdot))(\mathbf{x}_t,t)}{q_t(\mathbf{x}_t)/\beta(\mathbf{x}_t,t)} - \mathcal{L}\log(q_{\cdot}(\cdot)/\beta(\cdot,\cdot))(\mathbf{x}_t,t) \right] \mathrm{d}t.$$

To see that this is equivalent to $\mathcal{I}_{\mathrm{ISM}}$ and $\mathcal{I}_{\mathrm{DSM}}$, observe

$$\begin{aligned}
&\mathbb{E}_{q_t(\mathbf{x}_t)}\left[\frac{\mathcal{L}(q_\cdot(\cdot)/\beta(\cdot,\cdot))(\mathbf{x}_t,t)}{q_t(\mathbf{x}_t)/\beta(\mathbf{x}_t,t)} - \mathcal{L}\log(q_\cdot(\cdot)/\beta(\cdot,\cdot))(\mathbf{x}_t,t)\right] \\
&= \int_{\mathcal{X}} q_t(\mathbf{x}_t)\left\{\frac{\mathcal{L}(q_\cdot(\cdot)/\beta(\cdot,\cdot))(\mathbf{x}_t,t)}{q_t(\mathbf{x}_t)/\beta(\mathbf{x}_t,t)} - \mathcal{L}\log(q_\cdot(\cdot)/\beta(\cdot,\cdot))(\mathbf{x}_t,t)\right\}\mathrm{d}\nu(\mathbf{x}_t) \\
&= \int_{\mathcal{X}}\left\{\beta(\mathbf{x}_t,t)\hat{\mathcal{L}}\left(\frac{q_\cdot(\cdot)}{\beta(\cdot,\cdot)}\right) - q_t(\mathbf{x}_t)\hat{\mathcal{L}}\log\left(\frac{q_\cdot(\cdot)}{\beta(\cdot,\cdot)}\right)\right\}\mathrm{d}\nu(\mathbf{x}_t) \\
&= \int_{\mathcal{X}} q_t(\mathbf{x}_t)\left\{\frac{\hat{\mathcal{L}}^*\beta(\mathbf{x}_t,t)}{\beta(\mathbf{x}_t,t)} + \hat{\mathcal{L}}\log\beta(\mathbf{x}_t,t)\right\}\mathrm{d}\nu(\mathbf{x}_t) + const \\
&= \mathbb{E}_{q_t(\mathbf{x}_t)}\left[\frac{\hat{\mathcal{L}}^*\beta}{\beta} + \hat{\mathcal{L}}\log\beta\right] + const
\end{aligned}$$

and integrate from $t = 0$ to $t = T$.

F. Application to particular spaces

In this section, we show how our general framework can be applied in some particular cases of interest, namely to Euclidean diffusion processes, continuous-time Markov Chains on finite discrete state spaces, diffusions on Riemannian manifolds and the Wright–Fisher diffusion on the simplex.

A recurring theme we see in each example is that the default parameterisation given by our framework in terms of β is sub-optimal, either because we expect it to lead to numerical instabilities when optimising the training objective, or because it only captures a restricted subset of the class of reverse processes we are interested in. However, in each case it turns out to be possible to reparameterise the generative process in a way which captures a wider class of processes and lets us interpret the training objective on this wider class. This allows us to optimise our generative process over this wider class of processes. In addition this reparameterisation typically leads to a form of the objective that we expect to be more numerically stable in practice.

F.1. Real vector spaces

We show how our framework recovers the setup of Song et al. (2021), described in Section 2.1, in the case where $\mathcal{K}$ and $\mathcal{L}$ are the Euclidean diffusion processes given in Example 1. For convenience, we recall that X and Y satisfy the SDEs

$$\mathrm{d}X_t = \mu(X_t,t)\mathrm{d}t + \mathrm{d}\hat{B}_t, \qquad \mathrm{d}Y_t = b(Y_t,t)\mathrm{d}t + \mathrm{d}B_t, \tag{14}$$

respectively, and the corresponding generators are

$$\mathcal{K} = \partial_t + \mu\cdot\nabla + \frac{1}{2}\Delta, \qquad \mathcal{L} = \partial_t + b\cdot\nabla + \frac{1}{2}\Delta.$$

First, we check the assumptions made in Appendix B. If we let our reference measure ν be the Lebesgue measure, then Assumption 3 holds. Assumption 5 is satisfied whenever

b and μ are Lipschitz functions (Schilling and Partzsch, 2012, Corollaries 19.27 and 19.31), and Assumption 6 follows given the form of $\mathcal{K}$ above. For Assumption 7 we take $\mathcal{D}_0 = C_c^\infty(\mathbb{R}^d)$, the set of infinitely differentiable functions with compact support, and note that this is dense in $L^2(\mathcal{X}, \nu)$. Finally, we assume that the reverse process and p_0 are sufficiently regular that Assumptions 4, 8 and 9 hold.

Using integration by parts, we can calculate the adjoint of $\hat{\mathcal{K}}$. We have

$$\begin{aligned}\int f\hat{\mathcal{K}}h\mathrm{d}\nu &= \int f\left(\mu\cdot\nabla h + \frac{1}{2}\Delta h\right)\mathrm{d}\nu \\ &= -\int h\nabla\cdot(f\mu)\mathrm{d}\nu - \frac{1}{2}\int \nabla f\cdot\nabla h\mathrm{d}\nu \\ &= \int h\left(-\mu\cdot\nabla f - (\nabla\cdot\mu)f + \frac{1}{2}\Delta f\right)\mathrm{d}\nu,\end{aligned}$$

assuming f and h are sufficiently regular that all boundary terms are zero. Therefore,

$$\hat{\mathcal{K}}^* = -\mu\cdot\nabla - (\nabla\cdot\mu) + \frac{1}{2}\Delta.$$

We see that Assumption 1 holds if we let $c = -(\nabla\cdot\mu)$ and

$$\mathcal{M} = \partial_t - \mu\cdot\nabla + \frac{1}{2}\Delta,$$

noting that this is the generator of another diffusion process $\overline{Z}$ satisfying the SDE

$$\mathrm{d}Z_t = -\mu(Z_t, T-t)\mathrm{d}t + \mathrm{d}B'_t.$$

Given this form of $\mathcal{L}$ and $\mathcal{M}$, Assumption 2 then becomes

$$-\beta^{-1}\mu\cdot\nabla f + \frac{1}{2}\beta^{-1}\Delta f = b\cdot\nabla(\beta^{-1}f) + \frac{1}{2}\Delta(\beta^{-1}f) - fb\cdot\nabla(\beta^{-1}) - \frac{1}{2}f\Delta(\beta^{-1}),$$

which reduces to

$$\nabla\log\beta = \mu + b, \tag{15}$$

for some bounded measurable function β. This puts a restriction on the class of reverse processes $\mathcal{K}$ we may use; the condition that the drift μ must be expressible as $-b+\nabla\log\beta$ for some β is not automatically satisfied. However, the true time-reversal of the forward process will satisfy this property. In addition, we will show that we may reparameterise the training objective so that it can be interpreted for a broader class of reverse processes.

Assuming for the moment that Assumption 2 does hold, we can evaluate

$$\begin{aligned}\Phi(f) &= \frac{\mathcal{L}f}{f} - \mathcal{L}\log f \\ &= \frac{b\cdot\nabla f}{f} + \frac{1}{2}\frac{\Delta f}{f} - b\cdot\nabla\log f - \frac{1}{2}\Delta\log f \\ &= \frac{1}{2}\|\nabla\log f\|^2,\end{aligned}$$

and so the denoising score matching objective becomes

$$\mathcal{I}_{\mathrm{DSM}}(\beta) = \frac{1}{2}\int_0^T \mathbb{E}_{q_{0,t}(\mathbf{x}_0,\mathbf{x}_t)}\left[\left\|\nabla \log q_{t|0}(\mathbf{x}_t|\mathbf{x}_0) - \nabla \log \beta(\mathbf{x}_t,t)\right\|^2\right] \mathrm{d}t. \tag{16}$$

Looking at Equations (15) and (16) suggests that it is more natural to parameterise the reverse process in terms of $s_\theta(\mathbf{x},t) = \nabla \log \beta(\mathbf{x},t)$ instead of $\beta(\mathbf{x},t)$. Making this substitution, the objective becomes

$$\mathcal{I}_{\mathrm{DSM}}(\theta) = \frac{1}{2}\int_0^T \mathbb{E}_{q_{0,t}(\mathbf{x}_0,\mathbf{x}_t)}\left[\left\|\nabla \log q_{t|0}(\mathbf{x}_t|\mathbf{x}_0) - s_\theta(\mathbf{x}_t,t)\right\|^2\right] \mathrm{d}t,$$

recovering the objective of Song et al. (2021).

Parameterising in terms of $s_\theta(\mathbf{x},t)$ rather than $\beta(\mathbf{x},t)$ is preferable for a couple of reasons. First, $s_\theta(\mathbf{x},t)$ is targeting the score $\nabla \log q_t(\mathbf{x})$, while $\beta(\mathbf{x},t)$ is targeting $q_t(\mathbf{x})$, and we expect the former to typically be an easier target. Second, while Equation (16) only makes sense when the forward and backward processes are related via Assumption 2, the objective in Equation (3) is valid for any forward and backward diffusion processes as in Equation (14). Hence reparameterising allows us to capture a wider class of reverse processes in our optimisation.

F.2. Discrete state spaces

Next, we show how to apply our framework when X and Y are continuous-time Markov chains on a finite discrete state space as in Example 2. With a particular choice of parameterisation, we end up recovering the set-up of Campbell et al. (2022).

Recall that we start with $\mathcal{K} = \partial_t + A$ and $\mathcal{L} = \partial_t + B$, where A and B are the time-dependent generator matrices of X and Y respectively. From this it follows immediately that $\hat{\mathcal{K}}^* = A^T$. We will use the counting measure as our reference measure ν.

On a finite discrete space, all functions are bounded and have compact support, and $\mathcal{D}(\hat{\mathcal{K}}) = \mathcal{D}(\hat{\mathcal{L}}) = C_0(\mathcal{S})$ is the set of all functions on $\mathcal{X}$. Assumptions 3, 5, 6 and 7 follow immediately. In addition, we assume that the reverse process and p_0 are sufficiently regular that Assumptions 4, 8 and 9 always hold.

In order for Assumption 1 to hold, we need to find $\mathcal{M}$ and c such that $\mathcal{M}+c = \partial_t+\hat{\mathcal{K}}^*$ (viewed as operators). Since $\mathcal{M}$ should be the generator of another CTMC, we write $\mathcal{M} = \partial_t + D$ for some generator matrix D. We then require $D + c = A^T$, where c is viewed as a diagonal matrix and D must have zero row sums. This holds if and only if we take

$$c_{\mathbf{x}} = \sum_{\mathbf{y}\in\mathcal{X}} A_{\mathbf{yx}}, \qquad D_{\mathbf{xy}} = A_{\mathbf{yx}} - c_{\mathbf{x}}\mathbb{1}_{\mathbf{x}=\mathbf{y}}.$$

With this choice of $\mathcal{M}$, Assumption 2 becomes

$$\beta^{-1}(\mathbf{x},t)\sum_{\mathbf{z}\in\mathcal{X}} D_{\mathbf{xz}} f(\mathbf{z}) = \sum_{\mathbf{z}\in\mathcal{X}} B_{\mathbf{xz}}\, \beta^{-1}(\mathbf{z},t) f(\mathbf{z}) - f(\mathbf{x}) \sum_{\mathbf{z}\in\mathcal{X}} B_{\mathbf{xz}}\, \beta^{-1}(\mathbf{z},t)$$

for all $\mathbf{x} \in \mathcal{X}$. If we pick two distinct $\mathbf{x}, \mathbf{y}$ and set $f(\mathbf{z}) = \mathbb{1}_{\mathbf{z}=\mathbf{y}}$ in the above, we deduce

$$\beta^{-1}(\mathbf{x},t) D_{\mathbf{xy}} = \beta^{-1}(\mathbf{y},t) B_{\mathbf{xy}} \quad \text{for all } \mathbf{x} \neq \mathbf{y}.$$

Hence for Assumption 2 to hold, we require

$$A_{\mathbf{yx}} = \frac{\beta(\mathbf{x}, t)}{\beta(\mathbf{y}, t)} B_{\mathbf{xy}} \quad \text{for all } \mathbf{x} \neq \mathbf{y}. \tag{17}$$

An elementary check also shows that this condition is sufficient for Assumption 2 to hold for a given choice of β.

With this parameterisation, the implicit score matching objective becomes

$$\begin{aligned}\mathcal{I}_{\mathrm{ISM}}(\beta) &= \int_0^T \mathbb{E}_{q_t(\mathbf{x}_t)} \left[\frac{B^T \beta}{\beta} + B \log \beta \right] \mathrm{d}t \\ &= \int_0^T \mathbb{E}_{q_t(\mathbf{x}_t)} \left[\sum_{\mathbf{y} \in \mathcal{X}} \left\{ B_{\mathbf{y}\mathbf{x}_t} \frac{\beta(\mathbf{y}, t)}{\beta(\mathbf{x}_t, t)} + B_{\mathbf{x}_t \mathbf{y}} \log \frac{\beta(\mathbf{y}, t)}{\beta(\mathbf{x}_t, t)} \right\} \right] \mathrm{d}t.\end{aligned}$$

Unfortunately, fitting β directly using this objective is typically likely to perform poorly. This can be seen for a couple of reasons. Firstly, the optimal value of $\beta(\mathbf{x}, t)$ is $q_t(\mathbf{x})$, and so learning $\beta(\mathbf{x}, t)$ should be roughly as hard as targeting the marginals of the forward process directly. Secondly, the presence of β in the denominators can lead to numerical instabilities in regions where the forward process has low density.

Fortunately, we have at least a couple of methods for avoiding these problems available. The first is to find an equivalent formulation of the objective in terms of the generator of the reverse process, and then learn this generator using a denoising parameterisation. For $\mathbf{x} \neq \mathbf{y}$, we have

$$\begin{aligned} B_{\mathbf{xy}} \log \frac{\beta(\mathbf{y}, t)}{\beta(\mathbf{x}, t)} &= B_{\mathbf{xy}} \log \frac{B_{\mathbf{xy}}}{A_{\mathbf{yx}}} \\ &= -B_{\mathbf{xy}} \log A_{\mathbf{yx}} + const, \end{aligned}$$

where the constant depends only on the dynamics of the forward process, which are fixed. We can therefore write

$$\begin{aligned}\mathcal{I}_{\mathrm{ISM}}(A) &= \int_0^T \mathbb{E}_{q_t(\mathbf{x}_t)} \left[B_{\mathbf{x}_t \mathbf{x}_t} + \sum_{\mathbf{y} \neq \mathbf{x}_t} A_{\mathbf{x}_t \mathbf{y}} - \sum_{\mathbf{y} \neq \mathbf{x}_t} B_{\mathbf{x}_t \mathbf{y}} \log A_{\mathbf{y}\mathbf{x}_t} \right] \mathrm{d}t + const \\ &= \int_0^T \mathbb{E}_{q_t(\mathbf{x}_t)} \left[-A_{\mathbf{x}_t \mathbf{x}_t} - \sum_{\mathbf{y} \neq \mathbf{x}_t} B_{\mathbf{x}_t \mathbf{y}} \log A_{\mathbf{y}\mathbf{x}_t} \right] \mathrm{d}t + const,\end{aligned}$$

recovering the objective of Campbell et al. (2022). In addition, we can parameterise the reverse generator A via

$$A_{\mathbf{xy}}(\theta) = B_{\mathbf{yx}} \sum_{\mathbf{x}_0} \frac{q_{t|0}(\mathbf{y}|\mathbf{x}_0)}{q_{t|0}(\mathbf{x}|\mathbf{x}_0)} p_\theta^{(t)}(\mathbf{x}_0|\mathbf{x}_t) \quad \text{for } \mathbf{x} \neq \mathbf{y}, \tag{18}$$

where $p_\theta^{(t)}(\mathbf{x}_0|\mathbf{x}_t)$ is some learned estimate of the original datapoint $\mathbf{x}_0$ given the noised observation $\mathbf{x}_t$, and θ denotes the learnable parameters. This parameterisation should be

more stable, as it avoids potentially exploding denominators, and we expect predicting the original datapoint given the noised datapoint to be an easier goal than learning the marginals $q_t(\mathbf{x})$. See Campbell et al. (2022) for more details on this denoising parameterisation.

The second method is to reparameterise our objective in terms of the ratios $s_\theta(\mathbf{x}, \mathbf{y}, t) = \beta(\mathbf{y}, t)/\beta(\mathbf{x}, t)$. Doing this, the training objective becomes

$$\mathcal{I}_{\text{ISM}}(\theta) = \int_0^T \mathbb{E}_{q_t(\mathbf{x}_t)} \left[\sum_{\mathbf{y} \in \mathcal{X}} \left\{ B_{\mathbf{y}\mathbf{x}_t} s_\theta(\mathbf{x}_t, \mathbf{y}; t) - B_{\mathbf{x}_t \mathbf{y}} \log s_\theta(\mathbf{y}, \mathbf{x}_t; t) \right\} \right] \mathrm{d}t. \tag{19}$$

In addition, the generative process is now parameterised in terms of $s_\theta(\mathbf{x}, \mathbf{y}, t)$ via

$$A_{\mathbf{x}\mathbf{y}} = B_{\mathbf{y}\mathbf{x}} s_\theta(\mathbf{x}, \mathbf{y}; t) \quad \text{for } \mathbf{x} \neq \mathbf{y}. \tag{20}$$

Importantly, this objective matches the generalised objective from Section 3 when the noising and generative processes are related by Assumption 2, and is still minimised when $s_\theta(\mathbf{x}, \mathbf{y}; t) = q_t(\mathbf{y})/q_t(\mathbf{x})$.

This parameterisation is potentially beneficial for a couple of reasons. Firstly, by removing $\beta(\mathbf{x}, t)$ from the denominators, we expect that objective should be more numerically stable. Secondly, this parameterisation captures a wider class of potential reverse processes, since A is now given in terms of B via Equation (20), which is less restrictive than Equation (17).

As discussed further in Section 4, the integrand in Equation (19) may be viewed as a score matching objective for discrete state space. It shares certain similarities with ratio matching techniques (Hyvärinen, 2007), in particular targeting the ratios $\beta(\mathbf{y}, t)/\beta(\mathbf{x}, t)$. However, as far as we are aware this particular objective is not directly equivalent to any previously studied score matching objective in discrete state space (Hyvärinen, 2007; Lyu, 2009; Sohl-Dickstein et al., 2011).

F.3. Riemannian manifolds

Consider the case where $\mathcal{X}$ is a Riemannian manifold with metric tensor g and ν is the volume measure induced by g (so that Assumption 3 holds). A diffusion in $\mathcal{X}$ may be defined through its generator, so we let the noising and generative processes have generators

$$\mathcal{K} = \partial_t + \mu \cdot \nabla + \frac{1}{2}\Delta, \qquad \mathcal{L} = \partial_t + b \cdot \nabla + \frac{1}{2}\Delta.$$

respectively, where Δ is the Laplace-Beltrami operator defined in local coordinates by

$$\Delta f = \frac{1}{\sqrt{|g|}} \partial_i \big(\sqrt{|g|} g^{ij} \partial_j f \big)$$

and $|g|$ denotes the determinant of the metric tensor. For such processes, Assumption 5 is satisfied under mild regularity conditions on the manifold and the coefficients of the generators, as detailed by Molchanov (1968). As in the Euclidean diffusion case, Assumption 6 follows from the given form of $\mathcal{K}$, for Assumption 7 we may take $\mathcal{D}_0 = C_c^\infty(\mathcal{X})$ and note that this is dense in $L^2(\mathcal{X}, \nu)$ (Taylor, 2011, Section 4.4), and we

assume that the reverse process and p_0 are sufficiently regular that Assumptions 4, 8 and 9 hold.

To calculate the adjoint operator of $\hat{\mathcal{K}}$, we recall that the canonical volume element on $\mathcal{X}$ induced by g is given by

$$\mathrm{d}\omega = \sqrt{|g|}\,\mathrm{d}x^1 \wedge \cdots \wedge \mathrm{d}x^n$$

and the divergence of a vector field $a : \mathcal{X} \to T\mathcal{X}$ on a Riemannian manifold is given by

$$\nabla \cdot a = \frac{1}{\sqrt{|g|}} \partial_i (a^i \sqrt{|g|}).$$

Then, using the generalised Stokes' Theorem, we have

$$\begin{aligned}
\langle f, \mu \cdot \nabla h \rangle &= \int_{\mathcal{X}} f \mu^i \, (\partial_i h) \sqrt{|g|}\,\mathrm{d}x^1 \wedge \cdots \wedge \mathrm{d}x^n \\
&= -\int_{\mathcal{X}} h \, \partial_i (\mu^i f \sqrt{|g|})\,\mathrm{d}x^1 \wedge \cdots \wedge \mathrm{d}x^n \\
&= \langle -(\nabla \cdot \mu) f - (\mu \cdot \nabla f), h \rangle,
\end{aligned}$$

where we assume f and h are sufficiently smooth that we may disregard boundary terms. In addition, we have

$$\begin{aligned}
\langle f, \Delta h \rangle &= \int_{\mathcal{X}} f \partial_i (\sqrt{|g|} g^{ij} \partial_j h)\,\mathrm{d}x^1 \wedge \cdots \wedge \mathrm{d}x^n \\
&= -\int_{\mathcal{X}} \sqrt{|g|} g^{ij} (\partial_i f)(\partial_j h)\,\mathrm{d}x^1 \wedge \cdots \wedge \mathrm{d}x^n \\
&= \langle \Delta f, h \rangle.
\end{aligned}$$

We conclude that the adjoint operator is given by

$$\hat{\mathcal{K}}^* = -\mu \cdot \nabla - (\nabla \cdot \mu) + \frac{1}{2}\Delta.$$

Then, as in the Euclidean diffusion case we see that Assumption 1 holds if we let $c = -(\nabla \cdot \mu)$ and

$$\mathcal{M} = \partial_t - \mu \cdot \nabla + \frac{1}{2}\Delta,$$

noting that $\mathcal{M}$ is also the generator of a diffusion process Z on $\mathcal{X}$. We also find that Assumption 2 reduces to the condition $\nabla \log \beta = \mu + b$, as before.

Assuming this holds, we can evaluate

$$\begin{aligned}
\Phi(f) &= \frac{\mathcal{L}f}{f} - \mathcal{L}\log f \\
&= \frac{b \cdot \nabla f}{f} + \frac{1}{2}\frac{\Delta f}{f} - b \cdot \nabla \log f - \frac{1}{2}\Delta \log f \\
&= \frac{1}{2}\|\nabla \log f\|^2_{g(x)},
\end{aligned}$$

where $\|\cdot\|_{g(x)}$ denotes the norm on the tangent space $T_x\mathcal{X}$ induced by g.

Finally, as in the Euclidean diffusion we make a reparameterisation $s_\theta(\mathbf{x}, t) = \nabla \log \beta(\mathbf{x}, t)$ in order to sidestep Assumption 2 and provide an easier training target. The resulting denoising score matching objective is

$$\mathcal{I}_{\mathrm{DSM}}(\theta) = \frac{1}{2}\int_0^T \mathbb{E}_{q_{0,t}(\mathbf{x}_0,\mathbf{x}_t)}\left[\left\|\nabla \log q_{t|0}(\mathbf{x}_t|\mathbf{x}_0) - s_\theta(\mathbf{x}_t, t)\right\|^2_{g(\mathbf{x}_t)}\right] \mathrm{d}t,$$

which reproduces the result of De Bortoli et al. (2022) and Huang et al. (2022). Notably, we find that all the relevant formulae in the manifold case are essentially the same as in the Euclidean diffusion case, except for the inclusion of the metric tensor.

F.4. Wright–Fisher diffusions

Suppose we wish to approximate a distribution $p_{\mathrm{data}}(\cdot)$ over the space $\mathcal{X} = \mathcal{P}(E)$ of measures on a finite set $E = \{1, \ldots, N\}$. A natural class of stochastic processes on $\mathcal{X}$ are the Wright–Fisher diffusions, a model used in population genetics to describe the evolution of allele frequencies in a population over time (Ethier and Griffiths, 1993).

We can parameterise measures in $\mathcal{X}$ by tuples of real numbers $\mathbf{p} = (p_1, \ldots, p_N) \in [0,1]^N$ such that $\sum_{i=1}^N p_i = 1$. With this parameterisation, the Wright–Fisher diffusion has generator

$$\mathcal{L} = \partial_t + \frac{1}{2}\sum_{i,j=1}^N p_i\left(\delta_{ij} - p_j\right)\frac{\partial^2}{\partial p_i \partial p_j} + \sum_{i,j=1}^N q_{ij} p_i \frac{\partial}{\partial p_j},$$

with domain $\mathcal{D}(\mathcal{L}) = \{F|_{\mathcal{P}(E)} : F(p_1, \ldots, p_N) \in C^2(\mathbb{R}^N)\}$, where $(q_{ij})_{i,j=1,\ldots,N}$ is some matrix, potentially depending on $\mathbf{p}$ and t, such that $\sum_{j=1}^N q_{ij} = 0$ for each $i = 1, \ldots, N$.

If we take $q_{ij} = \frac{1}{2}\vartheta_j > 0$ for all $\mathbf{p} \in \mathcal{X}, t \in [0, T]$ and $i \neq j$, then this process is ergodic and its invariant distribution is Dirichlet(Θ), the Dirichlet distribution with parameters $\Theta = (\vartheta_1, \ldots, \vartheta_N)$ (Ethier and Griffiths, 1993). Moreover, the transition function of the process can be expressed as

$$P(t, \mathbf{p}, \cdot) = \sum_{n=0}^\infty d_n^\Theta(t) \sum_{\alpha \in (\mathbb{Z}_+^N) : |\alpha| = n} \binom{n}{\alpha} \prod_{i=1}^N p_i^{\alpha_i} \mathrm{Dirichlet}(\alpha + \Theta)(\cdot) \tag{21}$$

where $d_n^\Theta(t)$ are smooth functions of t given explicitly in Ethier and Griffiths (1993). It follows that if we take $\vartheta_j > 2$ for all j and we start the process in the interior of the simplex, then the process almost surely does not hit the boundary and the marginals of the forward process always vanish and have zero derivative at the boundary (since this holds for any Dirichlet distribution where all parameters are greater than 2).

Note that $\mathcal{X}$ is compact and hence locally compact and separable. Since we can view $\mathcal{X}$ as a subset of a linear subspace of $\mathbb{R}^N$, it also has a natural Lebesgue measure, which we take as the reference measure ν. Hence we satisfy Assumption 3.

We let our noising process have generator $\mathcal{L}$ as above and our generative process have generator

$$\mathcal{K} = \partial_t + \frac{1}{2}\sum_{i,j=1}^N p_i\left(\delta_{ij} - p_j\right)\frac{\partial^2}{\partial p_i \partial p_j} + \sum_{i,j=1}^N r_{ij} p_i \frac{\partial}{\partial p_j},$$

where $(r_{ij})_{i,j=1,\dots,N}$ is another matrix with zero row sums. The forward process Y is then Feller from Ethier and Kurtz (1993, Theorem 3.4). It follows that the extended forward process $\overline{Y}$ is also Feller and, since the process is pathwise continuous on a compact state space, this implies that the extended backward process $\overline{X}$ is also Feller, so Assumption 5 holds. Assumption 6 follows from the given form of $\mathcal{K}$, and for Assumption 7, we can take $\mathcal{D}_0 = \{F|_{\mathcal{P}(E)} : F(p_1, \dots, p_N) \in C^\infty(\mathbb{R}^N)\}$. As usual, we assume that the reverse process and p_0 are sufficiently regular that Assumptions 4, 8 and 9 hold.

In order to calculate the adjoint operator $\hat{\mathcal{K}}^*$, we require the following lemma, which is essentially a form of the integration by parts formula for the space $\mathcal{X}$.

LEMMA 5. *Suppose we have $F : \mathbb{R}^N \to \mathbb{R}^N$ such that for all $x \in \mathcal{X}$, $F(x) \cdot \mathbf{1} = 0$, where $\mathbf{1}$ is the unit vector in the $(1, \dots, 1)^T$ direction. In addition, suppose that $F(x) = 0$ for all $x \in \partial\mathcal{X}$. Then*

$$\int_{\mathcal{X}} \sum_{j=1}^{N} \frac{\partial F_j}{\partial p_j}(\mathbf{p}) \, \mathrm{d}\nu(\mathbf{p}) - \frac{1}{N} \int_{\mathcal{X}} \sum_{j,k=1}^{N} \frac{\partial F_k}{\partial p_j}(\mathbf{p}) \, \mathrm{d}\nu(\mathbf{p}) = 0.$$

PROOF. Since $F(x) \cdot \mathbf{1} = 0$ for all $x \in \mathcal{X}$, we can view F as a function from $\mathcal{X}$ to $T\mathcal{X}$, the tangent bundle of $\mathcal{X}$. Then, since $F(x) = 0$ for $x \in \partial\mathcal{X}$, by the generalised Stokes' theorem we have

$$\int_{\mathcal{X}} \nabla_{\mathcal{X}} \cdot F \, \mathrm{d}\nu = 0,$$

where $\nabla_{\mathcal{X}} \cdot F$ denotes the manifold divergence on $\mathcal{X}$. Finally, $\nabla_{\mathcal{X}} \cdot F = \nabla \cdot F - \mathbf{1} \cdot \nabla (F \cdot \mathbf{1})$, where ∇ is the standard gradient operator on $\mathbb{R}^N$, and so the result follows.

First, we need to calculate the adjoint of $\hat{\mathcal{K}}$. To deal with the first order term, we use Lemma 5 with $F_j = r_{ij} p_i f h$ for $i = 1, \dots, N$ in turn to get

$$\int_{\mathcal{X}} \sum_{j=1}^{N} \frac{\partial}{\partial p_j}(r_{ij} p_i f h) \, \mathrm{d}\nu(\mathbf{p}) - \frac{1}{N} \int_{\mathcal{X}} \sum_{j,k=1}^{N} \frac{\partial}{\partial p_j}(r_{ik} p_i f h) \, \mathrm{d}\nu(\mathbf{p}) = 0,$$

whenever $fh = 0$ on $\partial\mathcal{X}$. Since $\sum_{j=1}^{N} r_{ij} = 0$, the second term vanishes. Thus, summing over i we get

$$\int_{\mathcal{X}} \sum_{i,j=1}^{N} p_i f h \frac{\partial r_{ij}}{\partial p_j} \, \mathrm{d}\nu(\mathbf{p}) + \int_{\mathcal{X}} \sum_{i=1}^{N} r_{ii} f h \, \mathrm{d}\nu(\mathbf{p}) + \int_{\mathcal{X}} \sum_{i,j=1}^{N} r_{ij} p_i h \frac{\partial f}{\partial p_j} \, \mathrm{d}\nu(\mathbf{p}) + \int_{\mathcal{X}} \sum_{i,j=1}^{N} r_{ij} p_i f \frac{\partial h}{\partial p_j} \, \mathrm{d}\nu(\mathbf{p}) = 0,$$

from which we deduce that

$$\int_{\mathcal{X}} f \left(\sum_{i,j=1}^{N} r_{ij} p_i \frac{\partial}{\partial p_j} \right) h \, \mathrm{d}\nu(\mathbf{p}) = - \int_{\mathcal{X}} h \left(\sum_{i,j=1}^{N} p_i \frac{\partial r_{ij}}{\partial p_j} + \sum_{i=1}^{N} r_{ii} + \sum_{i,j=1}^{N} r_{ij} p_i \frac{\partial}{\partial p_j} \right) f \, \mathrm{d}\nu(\mathbf{p}) \tag{22}$$

whenever $fh = 0$ on $\partial\mathcal{X}$.

To deal with the second order term, we use Lemma 5 with $F_j = p_i(\delta_{ij} - p_j)(f\partial_i h)$ for each $i = 1, \ldots, N$ in turn to get

$$\int_{\mathcal{X}} \sum_{j=1}^{N} \frac{\partial}{\partial p_j}(p_i(\delta_{ij} - p_j)(f\partial_i h))\, \mathrm{d}\nu(\mathbf{p}) - \frac{1}{N}\int_{\mathcal{X}} \sum_{k,j=1}^{N} \frac{\partial}{\partial p_j}(p_i(\delta_{ik} - p_k)(f\partial_i h))\, \mathrm{d}\nu(\mathbf{p}) = 0,$$

whenever $f\partial_i h = 0$ on $\partial\mathcal{X}$ for each $i = 1, \ldots, N$. Expanding the LHS, we get

$$\int_{\mathcal{X}} \sum_{j=1}^{N} (\delta_{ij} - p_i - \delta_{ij}p_i)(f\partial_i h)\mathrm{d}\nu(\mathbf{p}) + \int_{\mathcal{X}} \sum_{j=1}^{N} p_i(\delta_{ij} - p_j)((\partial_j f)(\partial_i h) + f\partial_i\partial_j h)\mathrm{d}\nu(\mathbf{p})$$

$$- \frac{1}{N}\int_{\mathcal{X}} \sum_{j,k=1}^{N} (\delta_{ij}\delta_{ik} - \delta_{ij}p_k - \delta_{jk}p_i)(f\partial_i h)\mathrm{d}\nu(\mathbf{p}) - \frac{1}{N}\int_{\mathcal{X}} \sum_{j,k=1}^{N} p_i(\delta_{ik} - p_k)\partial_j(f\partial_i h)\mathrm{d}\nu(\mathbf{p}).$$

Now, the last term is zero since $\sum_{k=1}^{N} p_i(\delta_{ik} - p_k) = 0$. Simplifying and summing over i, we get

$$\int_{\mathcal{X}} \sum_{i=1}^{N} (1 - Np_i) f\partial_i h\, \mathrm{d}\nu(\mathbf{p}) + \int_{\mathcal{X}} f \left(\sum_{i,j=1}^{N} p_i(\delta_{ij} - p_j) \frac{\partial^2}{\partial p_i \partial p_j} \right) h\, \mathrm{d}\nu(\mathbf{p})$$

$$+ \int_{\mathcal{X}} \sum_{i,j=1}^{N} p_i(\delta_{ij} - p_j)(\partial_j f)(\partial_i h)\, \mathrm{d}\nu(\mathbf{p}) = 0.$$

By symmetry, we may reverse the roles of f and h in this last equation and subtract the resulting equations to get

$$\int_{\mathcal{X}} \sum_{i=1}^{N} (1 - Np_i) f\partial_i h\, \mathrm{d}\nu(\mathbf{p}) + \int_{\mathcal{X}} f \left(\sum_{i,j=1}^{N} p_i(\delta_{ij} - p_j) \frac{\partial^2}{\partial p_i \partial p_j} \right) h\, \mathrm{d}\nu(\mathbf{p})$$

$$= \int_{\mathcal{X}} \sum_{i=1}^{N} (1 - Np_i) h\partial_i f\, \mathrm{d}\nu(\mathbf{p}) + \int_{\mathcal{X}} h \left(\sum_{i,j=1}^{N} p_i(\delta_{ij} - p_j) \frac{\partial^2}{\partial p_i \partial p_j} \right) f\, \mathrm{d}\nu(\mathbf{p}) \qquad (23)$$

whenever $f\nabla h = h\nabla f = 0$ on $\partial\mathcal{X}$. Finally, applying Lemma 5 with $F_i = fh(1 - Np_i)$, we get

$$\int_{\mathcal{X}} \sum_{j=1}^{N} \frac{\partial}{\partial p_j}(fh(1 - Np_j))\, \mathrm{d}\nu(\mathbf{p}) - \frac{1}{N}\int_{\mathcal{X}} \sum_{i,j=1}^{N} \frac{\partial}{\partial p_j}(fh(1 - Np_i))\, \mathrm{d}\nu(\mathbf{p}) = 0.$$

whenever $fh = 0$ on $\partial\mathcal{X}$. Expanding, we have

$$\int_{\mathcal{X}} \sum_{j=1}^{N} h(1 - Np_j)\partial_j f\, \mathrm{d}\nu(\mathbf{p}) + \int_{\mathcal{X}} \sum_{j=1}^{N} f(1 - Np_j)\partial_j h\, \mathrm{d}\nu(\mathbf{p}) - N^2 \int_{\mathcal{X}} fh\mathrm{d}\nu(\mathbf{p})$$

$$- \frac{1}{N}\int_{\mathcal{X}} \sum_{i,j=1}^{N} (1 - Np_i)\partial_j(fh)\, \mathrm{d}\nu(\mathbf{p}) + N\int_{\mathcal{X}} fh\mathrm{d}\nu(\mathbf{p}) = 0,$$

which simplifies to

$$\int_{\mathcal{X}} \sum_{j=1}^{N} h(1-Np_j)\partial_j f \, \mathrm{d}\nu(\mathbf{p}) + \int_{\mathcal{X}} \sum_{j=1}^{N} f(1-Np_j)\partial_j h \, \mathrm{d}\nu(\mathbf{p}) - N(N-1)\int_{\mathcal{X}} fh\mathrm{d}\nu(\mathbf{p}) = 0. \tag{24}$$

Combining Equations (23) and (24), we see

$$\begin{aligned}\frac{1}{2}\int_{\mathcal{X}} f\left(\sum_{i,j=1}^{N} p_i(\delta_{ij}-p_j)\frac{\partial^2}{\partial p_i \partial p_j}\right) h \, \mathrm{d}\nu(\mathbf{p}) = \frac{1}{2}\int_{\mathcal{X}} h\left(\sum_{i,j=1}^{N} p_i(\delta_{ij}-p_j)\frac{\partial^2}{\partial p_i \partial p_j}\right) f \, \mathrm{d}\nu(\mathbf{p}) \\ + \int_{\mathcal{X}} h\left(\sum_{j=1}^{N}(1-Np_j)\frac{\partial}{\partial p_j}\right) f \, \mathrm{d}\nu(\mathbf{p}) \\ - \frac{N(N-1)}{2}\int_{\mathcal{X}} fh\mathrm{d}\nu(\mathbf{p}).\end{aligned} \tag{25}$$

Putting together Equations (22) and (25), we conclude that the operator

$$\begin{aligned}\mathcal{K}_0 = \frac{1}{2}\sum_{i,j=1}^{N} p_i\left(\delta_{ij}-p_j\right)\frac{\partial^2}{\partial p_i \partial p_j} + \sum_{j=1}^{N}(1-Np_j)\frac{\partial}{\partial p_j} - \frac{N(N-1)}{2} \\ - \sum_{i,j=1}^{N} p_i\frac{\partial r_{ij}}{\partial p_j} - \sum_{i=1}^{N} r_{ii} - \sum_{i,j=1}^{N} r_{ij}p_i\frac{\partial}{\partial p_j}\end{aligned}$$

satisfies $\langle \mathcal{K}_0 f, h\rangle = \langle f, \mathcal{K}h\rangle$ for all functions $f, h \in \{F|_{\mathcal{P}(E)} : F(p_1,\ldots,p_N) \in C^2(\mathbb{R}^N)\}$ such that $fh = f\nabla h = h\nabla f = 0$ on $\partial\mathcal{X}$. We conclude that $\hat{\mathcal{K}}^* = \mathcal{K}_0$ and $h \in \mathcal{D}(\hat{\mathcal{K}}^*)$ for all h such that $h = \nabla h = 0$ on $\partial\mathcal{X}$. Therefore, we choose to define

$$\mathcal{M} = \partial_t + \frac{1}{2}\sum_{i,j=1}^{N} p_i\left(\delta_{ij}-p_j\right)\frac{\partial^2}{\partial p_i \partial p_j} + \sum_{j=1}^{N}(1-Np_j)\frac{\partial}{\partial p_j} - \sum_{i,j=1}^{N} r_{ij}p_i\frac{\partial}{\partial p_j},$$

$$c = -\frac{N(N-1)}{2} - \sum_{i,j=1}^{N} p_i\frac{\partial r_{ij}}{\partial p_j} - \sum_{i=1}^{N} r_{ii}.$$

We see that Assumption 1 is satisfied, since v vanishes and has zero derivative on $\partial\mathcal{X}$ by our earlier remarks. Recalling that $q_{ij} = \frac{1}{2}\vartheta_j$ for $i \neq j$ and $\sum_{j=1}^{N} r_{ij} = 0$, if we let

$$u_{ij} = \vartheta_j + \frac{p_j}{p_i}(\vartheta_i - 1) - r_{ij}, \quad \text{for } i \neq j$$

and set $u_{ii} = -\sum_{j\neq i} u_{ij}$, then for each $j = 1, \dots, N$ we have

$$\begin{aligned}\sum_{i=1}^{N} u_{ij}p_i &= \sum_{i\neq j} u_{ij}p_i - \sum_{i\neq j} u_{ji}p_j \\ &= \sum_{i\neq j} (p_i\vartheta_j + p_j(\vartheta_i - 1)) - \sum_{i\neq j} (p_j\vartheta_i + p_i(\vartheta_j - 1)) - \sum_{i=1}^{N} r_{ij}p_i \\ &= 1 - Np_j - \sum_{i=1}^{N} r_{ij}p_i.\end{aligned}$$

We thus see that $\mathcal{M}$ is the generator of another Wright–Fisher process with transition matrix $(u_{ij})_{i,j=1,\dots,N}$. Hence Assumption 1 is satisfied. To check Assumption 2,

$$\begin{aligned}\beta\mathcal{L}(\beta^{-1}f) - \beta f\mathcal{L}(\beta^{-1}) &= \frac{\beta}{2}\sum_{i,j=1}^{N} p_i(\delta_{ij} - p_j)(\beta^{-1}(\partial_i\partial_j f) + 2(\partial_i f)(\partial_j\beta^{-1}) + f(\partial_i\partial_j\beta^{-1})) \\ &\quad + \beta\sum_{i,j=1}^{N} q_{ij}p_i(\beta^{-1}\partial_j f + f\partial_j\beta^{-1}) \\ &\quad - \frac{\beta f}{2}\sum_{i,j=1}^{N} p_i\left(\delta_{ij} - p_j\right)\partial_i\partial_j\beta^{-1} - \beta f\sum_{i,j=1}^{N} q_{ij}p_i\partial_j\beta^{-1} \\ &= \frac{1}{2}\sum_{i,j=1}^{N} p_i(\delta_{ij} - p_j)(\partial_i\partial_j f) - \sum_{i,j=1}^{N} p_i(\delta_{ij} - p_j)(\partial_i f)(\partial_j\log\beta) \\ &\quad + \sum_{i,j=1}^{N} q_{ij}p_i(\partial_j f).\end{aligned}$$

Thus Assumption 2 holds if and only if

$$\sum_{i=1}^{N} u_{ij}p_i = -\sum_{i=1}^{N} p_i(\delta_{ij} - p_j)(\partial_i\log\beta) + \sum_{i=1}^{N} q_{ij}p_i$$

for each $j = 1, \dots, N$. This is satisfied if we take

$$r_{ij} = \vartheta_j + \frac{p_j}{p_i}(\vartheta_i - 1) - p_j\frac{\partial(\log\beta)}{\partial p_i} - q_{ij} \tag{26}$$

for $i \neq j$ and $r_{ii} = -\sum_{j\neq i} r_{ij}$. We choose this parameterisation since if we start the forward process in its invariant distribution and learn β so that the generative process is the exact time reversal of the forward process then $\beta(\mathbf{p}, t) \propto q_t(\mathbf{x}) \propto \prod_{i=1}^{N} p_i^{\vartheta_i - 1}$. In this case, Equation (26) reduces to $r_{ij} = \vartheta_j - q_{ij}$, so this parameterisation ensures that if we start the forward process in its invariant distribution and learn the reverse process perfectly then the transition matrix $(r_{ij})_{i,j=1,\dots,n}$ we learn is equal to the transition matrix $(q_{ij})_{i,j=1,\dots,n}$ of the forward process.

We can then calculate the score matching operator $\Phi(f) = f^{-1}\mathcal{L}f - \mathcal{L}\log f$,

$$\begin{aligned}\Phi(f) &= \frac{1}{2}\sum_{i,j=1}^{N} p_i\left(\delta_{ij} - p_j\right) f^{-1}\frac{\partial^2 f}{\partial p_i \partial p_j} - \frac{1}{2}\sum_{i,j=1}^{N} p_i\left(\delta_{ij} - p_j\right)\frac{\partial^2(\log f)}{\partial p_i \partial p_j} \\ &\qquad + \sum_{i,j=1}^{N} q_{ij}p_i f^{-1}\frac{\partial f}{\partial p_j} - \sum_{i,j=1}^{N} q_{ij}p_i\frac{\partial(\log f)}{\partial p_j} \\ &= \frac{1}{2}\sum_{i,j=1}^{N} p_i\left(\delta_{ij} - p_j\right)\frac{\partial(\log f)}{\partial p_i}\frac{\partial(\log f)}{\partial p_j}.\end{aligned}$$

However, since we do not have access to the analytic forms of the transition kernel $q_{t|0}(\mathbf{p}_t|\mathbf{p}_0)$ for this model, we must fit β using the implicit score matching objective. We thus calculate

$$\begin{aligned}\frac{\hat{\mathcal{L}}^*\beta}{\beta} + \hat{\mathcal{L}}\log\beta &= \frac{1}{2}\sum_{i,j=1}^{N} p_i\left(\delta_{ij} - p_j\right)\beta^{-1}\frac{\partial^2\beta}{\partial p_i \partial p_j} + \sum_{j=1}^{N}(1 - Np_j)\beta^{-1}\frac{\partial\beta}{\partial p_j} - \frac{N(N-1)}{2} \\ &\quad - \sum_{i,j=1}^{N} p_i\frac{\partial q_{ij}}{\partial p_j} - \sum_{i=1}^{N} q_{ii} - \sum_{i,j=1}^{N} q_{ij}p_i\beta^{-1}\frac{\partial\beta}{\partial p_j} \\ &\quad + \frac{1}{2}\sum_{i,j=1}^{N} p_i\left(\delta_{ij} - p_j\right)\frac{\partial^2(\log\beta)}{\partial p_i \partial p_j} + \sum_{i,j=1}^{N} q_{ij}p_i\frac{\partial(\log\beta)}{\partial p_j} \\ &= \sum_{i,j=1}^{N} p_i\left(\delta_{ij} - p_j\right)\frac{\partial^2(\log\beta)}{\partial p_i \partial p_j} + \frac{1}{2}\sum_{i,j=1}^{N} p_i\left(\delta_{ij} - p_j\right)\frac{\partial(\log\beta)}{\partial p_i}\frac{(\partial\log\beta)}{\partial p_j} \\ &\quad + \sum_{j=1}^{N}(1 - Np_j)\frac{\partial(\log\beta)}{\partial p_j} + const,\end{aligned}$$

where we have discarded terms that do not depend on β. Noting that the loss and the reverse process only depend on β through $\partial(\log\beta)/\partial p_j$, we reparameterise in terms of $s^i_\theta(\mathbf{p}, t) = p_i\partial(\log\beta(\mathbf{p}, t))/\partial p_i$. (We include the extra factor of p_i for numerical stability reasons, since if we start in the stationary distribution then $p_i\partial(\log\beta(\mathbf{p}, t))/\partial p_i$ should be of constant scale.) Doing this, the implicit score matching objective becomes

$$\begin{aligned}\mathcal{I}_{\mathrm{ISM}}(\theta) = \int_0^T \mathbb{E}_{q_t(\mathbf{p}_t)}\Bigg[\sum_{i,j=1}^{N}\left(\delta_{ij} - p_j\right)\frac{\partial s^i_\theta(\mathbf{p}_t, t)}{\partial p_j} + \frac{1}{2}\sum_{i,j=1}^{N}(p_j^{-1}\delta_{ij} - 1)s^i_\theta(\mathbf{p}_t, t)s^j_\theta(\mathbf{p}_t, t) \\ + (1 - N)\sum_{j=1}^{N} s^j_\theta(\mathbf{p}_t, t)\Bigg]\mathrm{d}t, \quad (27)\end{aligned}$$

and the reverse process is parameterised as the Wright–Fisher diffusion with transition matrix $(r_{ij})_{i,j=1,\ldots,N}$ where

$$r_{ij}(\mathbf{p}, T - t) = \frac{1}{2}\vartheta_j + \frac{p_j}{p_i}(\theta_i - 1) - \frac{p_j}{p_i}s^i_\theta(\mathbf{p}, t), \quad i \neq j.$$

G. Proof of properties of the score matching operator

We give the proof of the properties of the score matching operator from Proposition 1.

PROPOSITION 1. *Let Y be a Feller process with semigroup operators $(Q_t)_{t\geq 0}$, generator $\mathcal{L}$ and associated score matching operator Φ. Then:*

(a) $\Phi(f) \geq 0$ for all f in the domain of Φ, with equality if f is constant;
(b) for any probability measures π_1, π_2 on $\mathcal{X}$ and $t \geq 0$,

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{KL}(\pi_1 Q_t||\pi_2 Q_t) = -\mathbb{E}_{\pi_1 Q_t}\left[\Phi\left(\frac{\mathrm{d}(\pi_1 Q_t)}{\mathrm{d}(\pi_2 Q_t)}\right)\right],$$

where $\mathrm{KL}(\pi_1 Q_t||\pi_2 Q_t)$ denotes the Kullback–Leibler divergence between $\pi_1 Q_t$, $\pi_2 Q_t$.

PROOF. Since log is a concave function, it follows that $\log(Q_t f) \geq Q_t(\log f)$ for all f in the domain of Φ with equality if f is constant. Hence

$$\frac{\log(Q_t f) - \log f}{t} \geq \frac{Q_t(\log f) - \log f}{t}$$

for all $t \geq 0$. Taking the limit $t \downarrow 0$, we deduce that $(\mathcal{L}f)/f \geq \mathcal{L}(\log f)$ which gives the first part of the lemma.

For the second part, we assume that $\pi_1 Q_t$ and $\pi_2 Q_t$ are absolutely continuous with respect to ν and let $\pi_{1,t}(\mathbf{x})$ and $\pi_{2,t}(\mathbf{x})$ respectively denote their densities. Then

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{KL}(\pi_1 Q_t||\pi_2 Q_t) &= \frac{\mathrm{d}}{\mathrm{d}t}\int \pi_{1,t}(\mathbf{x}) \log\left(\frac{\pi_{1,t}(\mathbf{x})}{\pi_{2,t}(\mathbf{x})}\right) \mathrm{d}\nu(\mathbf{x}) \\
&= \int \partial_t \pi_{1,t}(\mathbf{x}) \log\left(\frac{\pi_{1,t}(\mathbf{x})}{\pi_{2,t}(\mathbf{x})}\right) \mathrm{d}\nu(\mathbf{x}) + \int \pi_{1,t}(\mathbf{x}) \partial_t \log \pi_{1,t}(\mathbf{x}) \mathrm{d}\nu(\mathbf{x}) \\
&\quad - \int \pi_{1,t}(\mathbf{x}) \partial_t \log \pi_{2,t}(\mathbf{x}) \mathrm{d}\nu(\mathbf{x}) \\
&= \int \hat{\mathcal{L}}^* \pi_{1,t}(\mathbf{x}) \log\left(\frac{\pi_{1,t}(\mathbf{x})}{\pi_{2,t}(\mathbf{x})}\right) \mathrm{d}\nu(\mathbf{x}) + \int \hat{\mathcal{L}}^* \pi_{1,t}(\mathbf{x}) \mathrm{d}\nu(\mathbf{x}) \\
&\quad - \int \hat{\mathcal{L}}^* \pi_{2,t}(\mathbf{x}) \left(\frac{\pi_{1,t}(\mathbf{x})}{\pi_{2,t}(\mathbf{x})}\right) \mathrm{d}\nu(\mathbf{x})
\end{aligned}$$

using the Fokker–Planck equation on each term. Since $\langle \hat{\mathcal{L}}^* \pi_{1,t}, 1\rangle = \langle \pi_{1,t}, \hat{\mathcal{L}} 1\rangle = 0$, we may drop the second term and write

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{KL}(\pi_1 Q_t||\pi_2 Q_t) &= \mathbb{E}_{\pi_{1,t}(\mathbf{x})}\left[\hat{\mathcal{L}} \log\left(\frac{\pi_{1,t}(\mathbf{x})}{\pi_{2,t}(\mathbf{x})}\right)\right] - \mathbb{E}_{\pi_{1,t}(\mathbf{x})}\left[\left(\frac{\pi_{2,t}(\mathbf{x})}{\pi_{1,t}(\mathbf{x})}\right) \hat{\mathcal{L}} \left(\frac{\pi_{1,t}(\mathbf{x})}{\pi_{2,t}(\mathbf{x})}\right)\right] \\
&= -\mathbb{E}_{\pi_{1,t}(\mathbf{x})}\left[\Phi\left(\frac{\pi_{1,t}(\mathbf{x})}{\pi_{2,t}(\mathbf{x})}\right)\right],
\end{aligned}$$

which is the desired result.

H. Discrete-time approximation proofs

In this section, we give the proofs of Lemma 2 and Theorem 3 from Section 5. In order to prove Lemma 2, we use a couple of lemmas which we present first.

LEMMA 6. *Given processes $\overline{X}$ and $\overline{Y}$ as in Section 3, define a process $\overline{W}$ by setting $\overline{W}_t = (X_{T-t}, t)$ and denote its generator by $\mathcal{N}$. Then we have*

$$v\mathcal{N}g = (\mathcal{M} + c)(vg)$$

for all sufficiently rapidly decaying functions g.

PROOF. First, we let $\overleftarrow{\mathcal{K}} = -\partial_t + \hat{\mathcal{N}}$ denote the generator of the time-reversal of $\overline{X}$. Then, the integration by parts formula of Cattiaux et al. (2021) implies that for all sufficiently rapidly decaying test functions f and g we have

$$\langle p_t f, \mathcal{K}g + \overleftarrow{\mathcal{K}}g\rangle + \langle p_t, \Gamma(f,g)\rangle = 0,$$

where $\Gamma(f,g) = \mathcal{K}(fg) - f\mathcal{K}g - g\mathcal{K}f$ denotes the carré du champ operator associated to $\mathcal{K}$. We deduce that

$$\begin{aligned}
\langle f, p_t\overleftarrow{\mathcal{K}}g\rangle &= -\langle p_t f, \mathcal{K}g\rangle - \langle p_t, \mathcal{K}(fg) - f\mathcal{K}g - g\mathcal{K}f\rangle \\
&= -\langle p_t f, \partial_t g\rangle - \langle \hat{\mathcal{K}}^* p_t, fg\rangle + \langle \hat{\mathcal{K}}^*(gp_t), f\rangle \\
&= -\langle p_t \partial_t g, f\rangle - \langle g\partial_t p_t, f\rangle + \langle \hat{\mathcal{K}}^*(gp_t), f\rangle \\
&= \langle \hat{\mathcal{K}}^*(gp_t) - \partial_t(p_t g), f\rangle
\end{aligned}$$

where in the third line we have used the Fokker–Planck equation. Since f was arbitrary, it follows that

$$p_t\overleftarrow{\mathcal{K}}g = \hat{\mathcal{K}}^*(gp_t) - \partial_t(p_t g).$$

Finally if we substitute $t \mapsto T - t$ in this final equation, we get

$$v\mathcal{N}g = (\hat{\mathcal{K}}^* + \partial_t)(vg),$$

which gives the desired result when combined with the definition of $\mathcal{M}$ and c from Assumption 1.

LEMMA 7. *Suppose $\beta : \mathcal{S} \to (0,\infty)$ is a function such that Assumption 2 holds. If we define $\zeta = \beta^{-1}v$, then for any function f decaying sufficiently rapidly, ζ satisfies*

$$\zeta\mathcal{N}f = \mathcal{L}(f\zeta) - f\mathcal{L}\zeta.$$

PROOF. For any sufficiently rapidly decaying f satisfying $f\zeta \in \mathcal{D}(\mathcal{L})$ and $vf \in \mathcal{D}(\mathcal{M})$, using Lemma 6 we have

$$\begin{aligned}
\mathcal{L}(f\zeta) - f\mathcal{L}\zeta &= \mathcal{L}(v\beta^{-1}f) - f\mathcal{L}(\beta^{-1}v) \\
&= \beta^{-1}\mathcal{M}(vf) - \beta^{-1}f\mathcal{M}v \\
&= \beta^{-1}v\mathcal{N}f - c\beta^{-1}vf + c\beta^{-1}vf \\
&= \zeta\mathcal{N}f.
\end{aligned}$$

Now we can give the proofs of Lemma 2 and Theorem 3.

LEMMA 2. *Suppose X, Y are fixed generative and noising processes with marginals p, q as in Section 3, and suppose that they are related as in Assumptions 1 and 2 for some sufficiently regular function β. Then for any $0 < s < t < T$ with $\gamma = t - s$,*

$$\gamma\, \mathbb{E}_{q_s(\mathbf{x}_s)}\left[\frac{\hat{\mathcal{L}}^*\beta}{\beta} + \hat{\mathcal{L}}\log\beta\right] = \mathbb{E}_{q_{s,t}(\mathbf{x}_s,\mathbf{x}_t)}\left[\log\frac{q_{t|s}(\mathbf{x}_t|\mathbf{x}_s)}{p_{T-s|T-t}(\mathbf{x}_s|\mathbf{x}_t)}\right] + o(\gamma).$$

PROOF. Let $\mathbb{P}^{\mathbf{x}_s}$ and $\mathbb{Q}^{\mathbf{x}_s}$ denote the path measures of $\overline{W}$ and $\overline{Y}$ respectively on the interval $[s,t]$ when we condition on the initial value $\mathbf{x}_s$. Assuming β is sufficiently regular so that ζ is bounded away from zero and infinity and $\zeta^{-1}\mathcal{L}\zeta$ is bounded and continuous in the time variable, by Girsanov's theorem and Lemma 7 we have

$$\frac{\mathrm{d}\mathbb{P}^{\mathbf{x}_s}}{\mathrm{d}\mathbb{Q}^{\mathbf{x}_s}}(\omega) = \frac{\zeta(\omega_t,t)}{\zeta(\omega_s,s)}\exp\left\{-\int_s^t \frac{\mathcal{L}\zeta(\omega_\tau,\tau)}{\zeta(\omega_\tau,\tau)}\mathrm{d}\tau\right\}.$$

Taking logarithms and writing $\gamma = t - s$, to first order in γ for any fixed path ω this becomes

$$\log\frac{\mathrm{d}\mathbb{P}^{\mathbf{x}_s}}{\mathrm{d}\mathbb{Q}^{\mathbf{x}_s}}(\omega) = \log\frac{\zeta(\omega_t,t)}{\zeta(\omega_s,s)} - \gamma\frac{\mathcal{L}\zeta(\omega_s,s)}{\zeta(\omega_s,s)} + o(\gamma).$$

Since the first order terms depend only on the value of the path at its endpoints (ω_s,ω_t), we conclude that

$$\log\frac{q_{t|s}(\mathbf{x}_t|\mathbf{x}_s)}{p_{T-t|T-s}(\mathbf{x}_t|\mathbf{x}_s)} = -\log\frac{\zeta(\mathbf{x}_t,t)}{\zeta(\mathbf{x}_s,s)} + \gamma\frac{\mathcal{L}\zeta(\mathbf{x}_s,s)}{\zeta(\mathbf{x}_s,s)} + o(\gamma).$$

It follows that

$$\log\frac{q_{t|s}(\mathbf{x}_t|\mathbf{x}_s)}{p_{T-s|T-t}(\mathbf{x}_s|\mathbf{x}_t)} = \log\frac{v(\mathbf{x}_t,t)}{v(\mathbf{x}_s,s)} - \log\frac{\zeta(\mathbf{x}_t,t)}{\zeta(\mathbf{x}_s,s)} + \gamma\frac{\mathcal{L}\zeta(\mathbf{x}_s,s)}{\zeta(\mathbf{x}_s,s)} + o(\gamma).$$

Taking expectations and using the definition of the generator as a stochastic derivative, we have

$$\begin{aligned}\mathbb{E}_{\mathbb{Q}}\left[\log\frac{q_{t|s}(\mathbf{x}_t|\mathbf{x}_s)}{p_{T-s|T-t}(\mathbf{x}_s|\mathbf{x}_t)}\right] &= \gamma\mathbb{E}_{q_s(\mathbf{x}_s)}\left[\frac{\mathcal{L}\zeta(\mathbf{x}_s,s)}{\zeta(\mathbf{x}_s,s)} - \mathcal{L}\log\zeta(\mathbf{x}_s,s) + \mathcal{L}\log v(\mathbf{x}_s,s)\right] + o(\gamma)\\ &= \gamma\mathbb{E}_{q_s(\mathbf{x}_s)}\left[\frac{\hat{\mathcal{L}}^*\beta(\mathbf{x}_s,s)}{\beta(\mathbf{x}_s,s)} + \hat{\mathcal{L}}\log\beta(\mathbf{x}_s,s)\right] + o(\gamma),\end{aligned}$$

where in the final line we have used Lemma 1.

THEOREM 3. *For any DMM, the objective (11) for its natural discretisation is equivalent to the natural discretisation of $\mathcal{I}_{\mathrm{ISM}}$ to first order in $\overline{\gamma} = \max_{k=0,\dots,N-1}|t_{k+1} - t_k|$.*

PROOF. Given time steps $0 = t_0 < t_1 < \cdots < t_N = T$, define $\gamma_k = t_{k+1} - t_k$ for $k = 0,\dots,N-1$ and set $\overline{\gamma} = \max_{k=0,\dots,N-1}\gamma_k$. Then the natural discretisation of the objective $\mathcal{I}_{\mathrm{ISM}}(\beta)$ is given by

$$\sum_{k=0}^{N-1}\gamma_k\mathbb{E}_{q_{t_k}(\mathbf{x}_{t_k})}\left[\frac{\hat{\mathcal{L}}^*\beta}{\beta} + \hat{\mathcal{L}}\log\beta\right].$$

Using Lemma 2 with $s = t_{k-1}$ and $t = t_k$ for $k = 1, \ldots, N$, we get

$$\begin{aligned}
&\mathbb{E}_{\tilde{q}(\mathbf{x}_{t_{k-1}})}\left[\mathrm{KL}(\tilde{q}(\mathbf{x}_{t_{k-1}}|\mathbf{x}_{t_k})||\tilde{p}_\theta(\mathbf{x}_{t_{k-1}}|\mathbf{x}_{t_k}))\right] \\
&= \mathbb{E}_{\tilde{q}(\mathbf{x}_{t_{k-1}},\mathbf{x}_{t_k})}\left[\log \frac{\tilde{q}(\mathbf{x}_{t_k}|\mathbf{x}_{t_{k-1}})}{\tilde{p}_\theta(\mathbf{x}_{t_{k-1}}|\mathbf{x}_{t_k})}\right] + const \\
&= \gamma_{k-1}\, \mathbb{E}_{q_{t_{k-1}}(\mathbf{x}_{t_{k-1}})}\left[\frac{\hat{\mathcal{L}}^*\beta}{\beta} + \hat{\mathcal{L}}\log\beta\right] + o(\gamma_{k-1}) + const.
\end{aligned}$$

Putting this together, we see that

$$\mathrm{KL}(\tilde{q}(\mathbf{x}_{0:T})||\tilde{p}_\theta(\mathbf{x}_{0:T})) = \sum_{k=0}^{N-1} \gamma_k \mathbb{E}_{q_{t_k}(\mathbf{x}_{t_k})}\left[\frac{\hat{\mathcal{L}}^*\beta}{\beta} + \hat{\mathcal{L}}\log\beta\right] + o(\overline{\gamma}) + const,$$

so objective (11) is equivalent to the natural discretisation of $\mathcal{I}_{\mathrm{ISM}}$ to first order in $\overline{\gamma}$.

I. General equivalence between denoising autoencoders and score matching

A denoising autoencoder takes a datapoint $\mathbf{x}_0$ drawn from a data distribution q_0, noises it according to some density $q_\tau(\mathbf{x}_\tau|\mathbf{x}_0)$ and then tries to reconstruct $\mathbf{x}_0$ given the noised observation $\mathbf{x}_\tau$ (Vincent et al., 2008). Traditionally, $q_\tau(\mathbf{x}_\tau|\mathbf{x}_0)$ is taken to be Gaussian with mean $\mathbf{x}_0$ and some standard deviation σ and we make a point estimate $f_\theta(\mathbf{x}_\tau)$ for $\mathbf{x}_0$ given $\mathbf{x}_\tau$. The parameters θ are learned by minimising the MSE error

$$\mathcal{J}_{\mathrm{DAE}}(\theta) = \mathbb{E}_{q_{0,\tau}(\mathbf{x}_0,\mathbf{x}_\tau)}\left[\|f_\theta(\mathbf{x}_\tau) - \mathbf{x}_0\|^2\right].$$

For a general denoising autoencoder on state space $\mathcal{X}$, we allow a probabilistic reconstruction $p^{(\theta)}_{0|\tau}(\mathbf{x}_0|\mathbf{x}_\tau)$ of $\mathbf{x}_0$ depending on a set of parameters θ, rather than a point estimate. We fit θ by minimising the objective

$$\mathcal{J}_{\mathrm{DAE}}(\theta) = \mathbb{E}_{q_{0,\tau}(\mathbf{x}_0,\mathbf{x}_\tau)}\left[-\log p^{(\theta)}_{0|\tau}(\mathbf{x}_0|\mathbf{x}_\tau)\right].$$

Note that this reduces to the MSE objective in the case where $\mathcal{X} = \mathbb{R}^d$ and $p^{(\theta)}_{0|\tau}(\mathbf{x}_0|\mathbf{x}_\tau)$ is Gaussian with mean $f_\theta(\mathbf{x}_\tau)$.

Suppose now that we have a generalised denoising autodencoder where the noising distribution $q_{0,\tau}(\mathbf{x}_0, \mathbf{x}_\tau)$ is given by the endpoints of a Markov process on $\mathcal{X}$ with generator $\mathcal{L}$ and the denoising distribution $p^{(\theta)}_{0|\tau}(\mathbf{x}_0|\mathbf{x}_\tau)$ is given by the endpoints of a Markov process on $\mathcal{X}$ with generator $\mathcal{K}$. Suppose further that we parameterise the denoising process $\mathcal{K}$ via some function $\beta(\mathbf{x}, t)$ according to Assumptions 1 and 2 as in Section 3. Then Lemma 2 implies that $\mathcal{J}_{\mathrm{DAE}}$ is equivalent to first order to the objective

$$\mathcal{J}_{\mathrm{ISM}}(\beta) = \mathbb{E}_{q_\tau(\mathbf{x}_\tau)}\left[\frac{\hat{\mathcal{L}}^*\beta(\mathbf{x}_\tau,\tau)}{\beta(\mathbf{x}_\tau,\tau)} + \hat{\mathcal{L}}\log\beta(\mathbf{x}_\tau,\tau)\right],$$

or alternatively to the corresponding generalised denoising score matching objective as in Section 4.

This generalises the result of Vincent (2011), which demonstrated an equivalence between denoising autoencoders and denoising score matching in the case of Gaussian noise on $\mathbb{R}^d$. Indeed, we recover their result by considering the case where $q_{\tau|0}(\mathbf{x}_\tau|\mathbf{x}_0)$ and $p^{(\theta)}_{0|\tau}(\mathbf{x}_0|\mathbf{x}_\tau)$ are Gaussian, noting that these distributions are naturally induced as the distributions of the endpoints of diffusion processes.

Our work extends this equivalence between denoising autoencoders and generalised score matching as described in Section 4 to arbitrary state spaces and noising/denoising distributions, provided that the noising and denoising distributions can be viewed as the marginals at the endpoints of Markov processes with known generators.

J. Experimental details

We give the details of our experimental set-up and results from Section 6. Code for all of our experiments can be found at `github.com/yuyang-shi/generalized-diffusion`.

J.1. Inference on $\mathbb{R}^d$ using diffusion processes

The g-and-k distribution with parameters (A, B, g, k) is defined via its quantile function

$$F^{-1}(q|A, B, g, k) = A + B\left[1 + 0.8\tanh\left(\frac{gz(q)}{2}\right)\right]\left(1 + z(q)^2\right)^k z(q),$$

where $z(q)$ denotes the qth quantile of the standard Gaussian distribution, and we require $B > 0$ and $k > -0.5$. The parameters A, B, g, k control the location, scale, skewness and kurtosis of the distribution respectively (Prangle, 2020). The prior on the parameters is uniform on $[0, 10]^4$. For the diffusion model, we centre and rescale each parameter linearly to $[-1, 1]$ in our implementation, and transform back to $[0, 10]$ for reporting.

As our noising process, we use the Ornstein–Uhlenbeck process $\mathrm{d}Y_t = -\frac{1}{2}Y_t\mathrm{d}t + \mathrm{d}B_t$. This has generator $\mathcal{L} = \partial_t - \frac{1}{2}\mathbf{x}\cdot\nabla + \frac{1}{2}\Delta$ and transition densities $q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$ which are Gaussian and available analytically. We can sample from the forward process at time t by sampling $\mathbf{x}_0 \sim q_0(\mathbf{x}_0)$ and then $\mathbf{x}_t \sim q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$. In practice, we apply a time-rescaling to the noising process following Song et al. (2021), in order to apply less noise at small times and move more quickly to the reference distribution at large times, by considering

$$\mathrm{d}Y_t = -\frac{1}{2}\beta(t)Y_t\mathrm{d}t + \sqrt{\beta(t)}\mathrm{d}B_t.$$

The β schedule is set to be linear and monotonically increasing, i.e.

$$\beta(t) = \beta_{\min} + (\beta_{\max} - \beta_{\min})t. \tag{28}$$

We set $\beta_{\min} = 0.001$ and $\beta_{\max}$ is selected using a grid search from $2, 4, 6, 8, 10$.

The reverse process is parameterised in terms of a conditional score network $s_\theta(\mathbf{x}_t, \boldsymbol{\xi}, t)$ using multilayer perceptrons (MLPs). We first encode $\mathbf{x}$ and $\boldsymbol{\xi}$ into 128-dimensional encodings using two separate MLPs with 3 layers and 512 hidden units in each layer. We then concatenate the two encodings as well as the time t and pass through another MLP with 3 layers and 512 hidden units in each layer. The total number of neural

network parameters is approximately 1.9M. For $N = 250$, we take in $\boldsymbol{\xi}$ the full set of order statistics as inputs to our network, i.e. we sort the observation $\boldsymbol{\xi}$ and take all $n = 250$ values. For $N = 10000$, we take $n = 100$ evenly-spaced order statistics from our observation as inputs, following Fearnhead and Prangle (2012).

Since we have access to the analytic transition densities, we train using the denoising score matching objective $\mathcal{I}_{\mathrm{DSM}}(\theta)$. We use a total of 10^6 training samples $(\mathbf{x}_0, \boldsymbol{\xi}_0) \sim p_{\mathrm{data}}$ during training. We optimise the network using the Adam optimiser with batch size 512 and learning rate 0.0001 with a cosine annealing schedule for 2.5M iterations. For sampling, we use the Euler-Maruyama method with 1000 steps to simulate from the reverse SDE.

The ground truth posterior density is estimated with MCMC samples generated using the R package `gk` (Prangle, 2020). We compare our method with the semi-automatic ABC (SA-ABC) and Wasserstein SMC (W-SMC) methodologies using the R packages `abctools` (Nunes and Prangle, 2015) and `winference` (Bernton et al., 2019), as well as with Sequential Neural Posterior (Greenberg et al., 2019), Likelihood (Papamakarios et al., 2019) and Ratio Estimation (Durkan et al., 2020) approaches (SNPE, SNLE and SNRE) using the `sbi` Python package (Tejero-Cantero et al., 2020). All methods are set to use 10^6 data samples to generate 5000 posterior samples. We note that the default configurations offered by the `sbi` package for SNPE, SNLE and SNRE use comparatively smaller neural networks compared to our choice of score network $s_\theta(\mathbf{x}_t, \boldsymbol{\xi}, t)$ detailed above. We have correspondingly increased the size of the neural networks for the three methods to approximately the same number of parameters. We also use Neural Spline Flows (NSFs, Durkan et al. (2019)) for SNPE as it is reported to have superior performance (Lueckmann et al., 2021). Other settings are kept to the default values.

Compared to SA-ABC and W-SMC methodologies, neural-network based approaches including our DMM model require fitting a neural network and therefore are more computationally expensive at training time. However, our model is able to produce more accurate posterior estimates for fixed $\boldsymbol{\xi}_0$, and perform amortised inference across a range of parameter values using the same number of 10^6 data samples. Therefore, it is comparatively more data-efficient.

As well as the plots in the main text, we also provide a pair plot comparing the approximate posterior from our diffusion model to the ground truth joint distribution in Fig. 8. We see that our model provides results very close to the ground truth for the parameters A, B and g and can model the dependency between parameters, but gives a wider estimate in its reproduction of the posterior over k.

J.2. MNIST digit image inpainting using discrete-space CTMCs

Our implementation in discrete space closely follows that of Campbell et al. (2022), and we refer to their paper for further details. We denote our states as $\mathbf{x}_0 = (\mathbf{x}_0^1 \ldots, \mathbf{x}_0^D)$ and for our noising process we use a CTMC with generator matrix $B := B^{1:D}(\mathbf{x}^{1:D}, \mathbf{y}^{1:D})$ which factorises over the dimensions, so $B^{1:D}(\mathbf{x}^{1:D}, \mathbf{y}^{1:D}) = \sum_{i=1}^{D} \tilde{B}(\mathbf{x}^i, \mathbf{y}^i) \mathbb{1}_{\mathbf{x}^{1:D\setminus i} = \mathbf{y}^{1:D\setminus i}}$ for some rate matrix $\tilde{B}$ acting on a single dimension. Thus each pixel evolves independently as a CTMC on $\{0, \ldots, 255\}$ with rate matrix $\tilde{B}$. We use the Gaussian rate matrix of Campbell et al. (2022) for $\tilde{B}$, which respects the ordinal structure of our state space

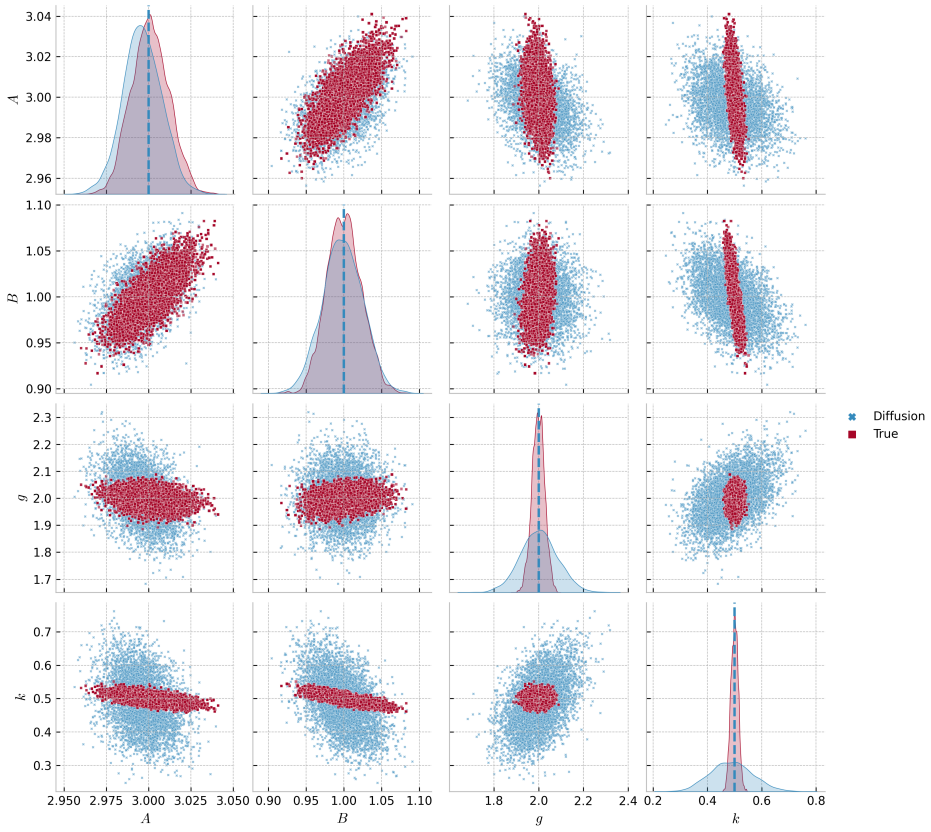

Fig. 8. Pair plots of the simulated posterior samples from the diffusion model and the ground truth distribution using MCMC for the g-and-k distribution example, with $\mathbf{x}_{\text{true}} = (3, 1, 2, 0.5)$ and $N = 10000$. The off-diagonal plots are the pairwise scatter plots between each component of $\mathbf{x}$, and the diagonal plots reproduce each parameter's marginal kernel density estimate.

and has a discretised Gaussian as its invariant distribution. The transition probabilities for this forward process can be calculated analytically efficiently by diagonalising the matrix and using matrix exponentials. This allows us to sample directly from the forward process at time t.

Since we have access to the forward transition probabilities, we use the denoising parameterisation of the reverse process in terms of $p_\theta^{(t)}(\mathbf{x}_0|\mathbf{x}_t)$ given in Equation (18), which we expect to lead to more stable training. We parameterise $p_\theta^{(t)}(\mathbf{x}_0|\mathbf{x}_t, \boldsymbol{\xi})$ using a convolutional U-net (Ho et al., 2020), taking as inputs both $\mathbf{x}_t$ and $\boldsymbol{\xi}$ (concatenated in the channel dimension), as well as a sinusoidal embedding of the time t. The total number of neural network parameters is approximately 6.1M. The output of the network is defined as the mean and log scale of a logistic distribution for each pixel. The logistic distribution is then discretised into bins $\{0, \ldots, 255\}$, and $p_\theta^{(t)}(\mathbf{x}_0|\mathbf{x}_t, \boldsymbol{\xi})$ is defined as the product of the discretised logistic distributions across dimensions.

We used the MNIST dataset (LeCun et al., 2010) which consists of images of hand-

Table 1. PSNR and SSIM scores for MNIST 14x14 inpainting using discrete-space and continuous-space DMMs. Higher values denote better performance.

	Discrete-space	Continuous-space (raw)	Continuous-space (rounded)
PSNR	16.63	16.72	**16.75**
SSIM	**0.757**	0.706	0.723

written digits. To train our model, we minimise the objective given in Example 6. For optimisation, we use the Adam optimiser with batch size 128 and learning rate 0.0002 for 1M iterations. In order to simulate the reverse process efficiently, we use a tau-leaping approximation with 1000 steps (for more details see Campbell et al. (2022)).

We compare our method to a continuous state space approach, as used for example in Song et al. (2021) and presented in Appendix F.1. We first normalize the data to range $[-1, 1]$, and then learn a continuous-space diffusion model with an Ornstein—Uhlenbeck noising process. All training configurations are kept the same as the discrete-space DMM. We report the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) for both methods in Table 1. PSNR and SSIM are two image quality metrics which measure the similarity between the generated posterior image and the ground truth. PSNR measures the pixel-by-pixel difference between two images and is a direct transformation of the mean squared error (MSE), whereas SSIM is a structural and more perceptional metric based on luminance, contrast and structure. For the continuous-space diffusion model, we report values for both the raw output samples (rescaled back to original scale), as well as with a further rounding step to the nearest integer in $\{0, \dots, 255\}$. The discrete-space and continuous-space models appear to achieve comparable results, with the discrete-space model having a slightly worse PSNR score, but slightly better SSIM score, suggesting comparable perceptual quality.

J.3. Large-scale image super-resolution using discrete-space CTMCs

We perform an additional experiment using discrete-space DMMs for a large-scale image inverse problem on the ImageNet dataset (Russakovsky et al., 2015). We train a DMM using CTMC noising and generative processes to perform 4-fold image super-resolution.

Each input image has 64×64 pixels and three RGB colour channels, and we aim to output images at the higher resolution of 256×256 pixels which are consistent with the input images. Our state space $\mathcal{X} = \{0, \dots, 255\}^{3 \times 256 \times 256}$.

The noising process, reverse process parameterisation, and neural network design are the same as in Section J.2, but we use a larger neural network for this task. As the starting point of our network optimisation, we utilise the pretrained network weights for continuous diffusions by Dhariwal and Nichol (2021), but we retrain the network for our discrete-space DMM using the objective in Example 6. The total number of neural network parameters is approximately 311.8M. We train the network using the Adam optimiser with batch size 4 and learning rate 2×10^{-5} for an additional 200000 iterations. For sampling, we use tau-leaping with 1000 steps.

We plot the simulated super-resolution samples in Fig. 9 for a number of low-resolution images generated from the ImageNet validation dataset. As shown in the images, the discrete diffusion model outputs different super-resolution samples that are realistic to

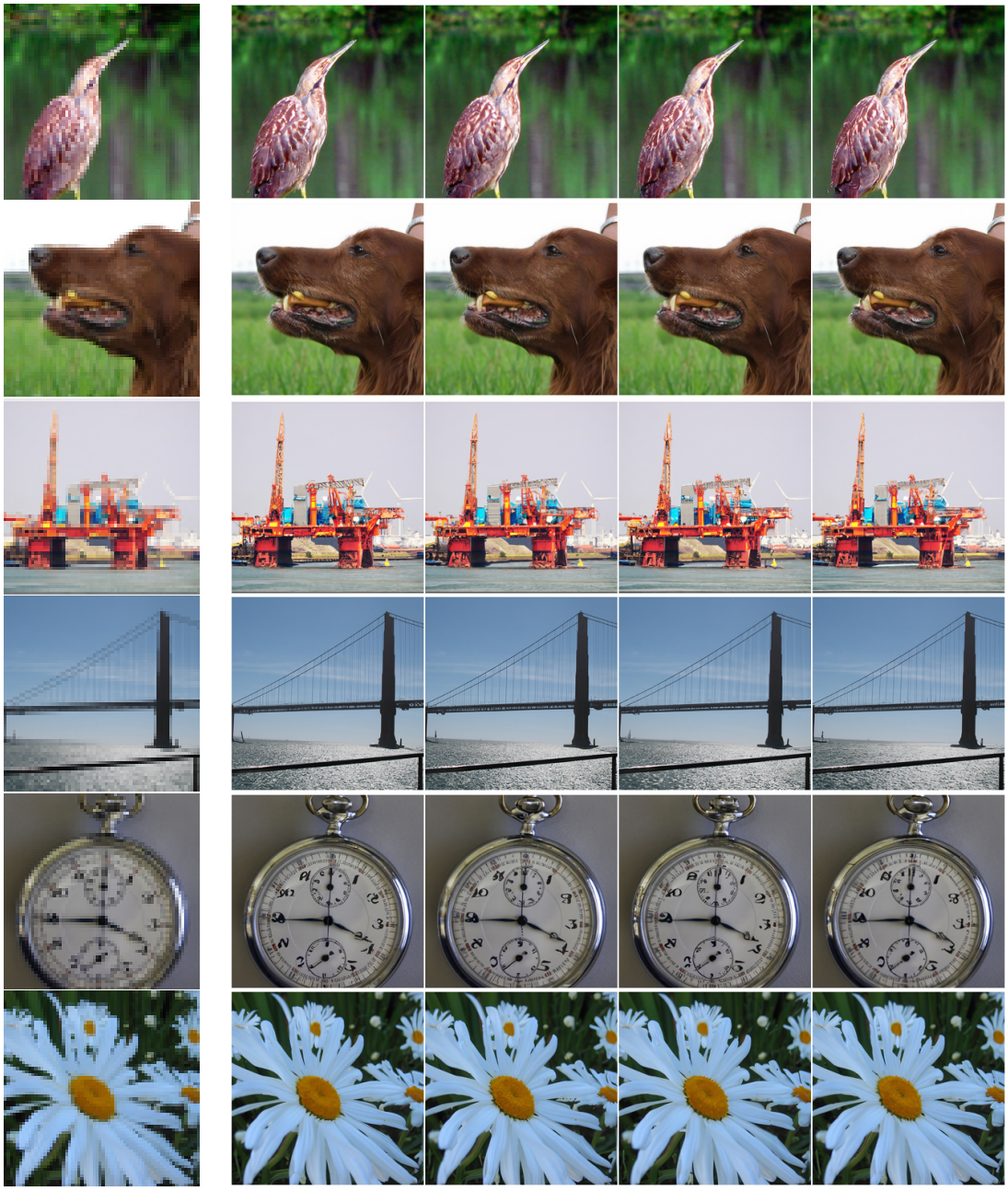

Fig. 9. Image super-resolution results ($64 \times 64 \rightarrow 256 \times 256$) on the ImageNet dataset using our DMM. The first column is the input image and remaining columns are samples from the DMM.

the eye, and coherent with the low-resolution images, demonstrating that DMMs can continue to provide high-quality posterior samples even in very high-dimensional scenarios situations where the prior $p_{\text{data}}(\mathbf{x})$ is unavailable and standard ABC or MCMC techniques are not available.

J.4. Modelling distributions on $SO(3)$ using manifold diffusions

Recall that our noising process on $SO(3)$ is Brownian motion with generator $\mathcal{L} = \partial_t + \frac{1}{2}\Delta$. Since $SO(3)$ is compact, this converges to the uniform measure for large times; see e.g. De Bortoli et al. (2022). For this process, the transition probabilities can be explicitly written as

$$q_{t|0}(\mathbf{x}_t|\mathbf{x}_0) \propto \sum_{\ell=0}^{\infty}(2\ell+1)e^{-\ell(\ell+1)t/2}\frac{\sin\left(\left(\ell+\frac{1}{2}\right)\alpha\right)}{\sin(\alpha/2)}, \tag{29}$$

where $\alpha = \arccos\left[2^{-1}(\text{Tr}(\mathbf{x}_0^T\mathbf{x}_t)-1)\right]$ is the angle between $\mathbf{x}_t$ and $\mathbf{x}_0$, and $\mathbf{x}_t, \mathbf{x}_0 \in SO(3)$ are in matrix form. For completeness, we provide the derivation of this result below in Section J.4.1.

Given this expression, to sample from $q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$, we follow Leach et al. (2022) and first sample the rotation axis v uniformly from the sphere $S^2 \subset \mathbb{R}^3$. Then, we sample the rotation angle $\alpha \in [0,\pi]$ using inverse transform sampling from the distribution

$$f_t(\alpha) = \frac{1-\cos(\alpha)}{\pi}\sum_{\ell=0}^{\infty}(2\ell+1)e^{-\ell(\ell+1)t/2}\frac{\sin\left(\left(\ell+\frac{1}{2}\right)\alpha\right)}{\sin(\alpha/2)},$$

where the normalising factor $(1-\cos(\alpha))/\pi$ is the measure on rotation angles induced by the uniform measure on $SO(3)$. For larger t, we find that the above series converges quickly and evaluating summation terms up to $l = 5$ gives an accurate approximation. For $t < 1$, the above series converges slowly, and so we use the approximation

$$f_t(\alpha) \approx \frac{1-\cos(\alpha)}{2\sqrt{\pi}\sin(\alpha/2)}\left(\frac{t}{2}\right)^{-\frac{3}{2}} e^{\frac{t}{8}-\frac{\alpha^2}{2t}}\left[\alpha - e^{-\frac{2\pi^2}{t}}\left((\alpha-2\pi)e^{\frac{2\pi\alpha}{t}} + (\alpha+2\pi)e^{-\frac{2\pi\alpha}{t}}\right)\right]$$

from Leach et al. (2022) instead. From the angle α and the axis $v = (x, y, z)$, we define the skew symmetric matrix V associated to v to be

$$V = \begin{pmatrix} 0 & z & -y \\ -z & 0 & x \\ y & -x & 0 \end{pmatrix}$$

and calculate the corresponding rotation matrix using Rodrigues' formula

$$R = I + \sin(\alpha)V + (1-\cos(\alpha))V^2.$$

Finally, we set $\mathbf{x}_t = R\mathbf{x}_0$. In this way, we can directly sample from the noising process at time t.

The reverse process is generated by $\mathcal{K} = \partial_t + s_\theta(\mathbf{x},t)\cdot\nabla + \frac{1}{2}\Delta$ by Example 7, and the score network is parameterised as $s_\theta(\mathbf{x},t) = \sum_{i=1}^3 s_\theta^i(\mathbf{x},t)E_i(\mathbf{x})$, using a basis $\{E_i\}_{i=1}^3$ of the tangent bundle.

We use the denoising score matching objective $\mathcal{I}_{\mathrm{DSM}}(\theta)$ to learn θ (see Section F.3). To compute the score $\nabla \log q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$, we use automatic differentiation on Equation (29), where $\mathbf{x}_t, \mathbf{x}_0 \in \mathbb{R}^{3\times 3}$ are represented in matrix form, followed by projection to the tangent space at $\mathbf{x}_t$. For small times, we find this can be numerically unstable, and so we use Varadhan's approximation

$$\lim_{t\to 0}\ t\nabla \log q_{t|0}(\mathbf{x}_t|\mathbf{x}_0) = \exp_{\mathbf{x}_t}^{-1}(\mathbf{x}_0)$$

for the heat kernel $q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$ at small times instead (De Bortoli et al., 2022).

Once we have learned the score network, we generate approximate samples from the reverse process using the Geodesic Random Walk method of De Bortoli et al. (2022), which corresponds to performing an Euler-Maruyama discretisation, taking Gaussian steps in the tangent space and then projecting back to the manifold using the exponential map.

J.4.1. Derivation of analytic transition probabilities

First, we calculate the metric tensor using the quaternion chart on $SO(3)$, where the unit quaternion $w + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$ represents a rotation by an angle $\alpha = 2\cos^{-1}(w)$ about the axis (x, y, z), and we consider the coordinates (x, y, z) to be our local chart. If $r = w + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$, we find the metric at r by considering two small displacements $r + \mathrm{d}r$ and $r + \mathrm{d}r'$, rotating r back to the identity, and then using the fact that near the identity the metric is given by $4\mathrm{d}x^2 + 4\mathrm{d}y^2 + 4\mathrm{d}z^2$ (where the scaling is chosen to correspond to the definition of the exponential map used by De Bortoli et al. (2022) and Leach et al. (2022)). Writing

$$\begin{aligned} r + \mathrm{d}r &= (w + \mathrm{d}w) + (x + \mathrm{d}x)\mathbf{i} + (y + \mathrm{d}y)\mathbf{j} + (z + \mathrm{d}z)\mathbf{k}, \\ r + \mathrm{d}r' &= (w + \mathrm{d}w') + (x + \mathrm{d}x')\mathbf{i} + (y + \mathrm{d}y')\mathbf{j} + (z + \mathrm{d}z')\mathbf{k}, \end{aligned}$$

where we have $w\mathrm{d}w + x\mathrm{d}x + y\mathrm{d}y + z\mathrm{d}z = 0$ and $w\mathrm{d}w' + x\mathrm{d}x' + y\mathrm{d}y' + z\mathrm{d}z' = 0$, and noting that composition of rotations corresponds to multiplication in the quaternion algebra, we have

$$\begin{aligned} r^{-1}(r + \mathrm{d}r) &= (w - x\mathbf{i} - y\mathbf{j} - z\mathbf{k})\left((w + \mathrm{d}w) + (x + \mathrm{d}x)\mathbf{i} + (y + \mathrm{d}y)\mathbf{j} + (z + \mathrm{d}z)\mathbf{k}\right) \\ &= 1 + (-x\mathrm{d}w + w\mathrm{d}x - y\mathrm{d}z + z\mathrm{d}y)\,\mathbf{i} + (-y\mathrm{d}w + w\mathrm{d}y - z\mathrm{d}x + x\mathrm{d}z)\,\mathbf{j} \\ &\quad + (-z\mathrm{d}w + w\mathrm{d}z - x\mathrm{d}y + y\mathrm{d}x)\,\mathbf{k} \end{aligned}$$

and similarly for $r^{-1}(r + \mathrm{d}r')$. Therefore, the metric is expressed by

$$4\left\{\left(w + \frac{x^2}{w}\right)\mathrm{d}x + \left(-y + \frac{xz}{w}\right)\mathrm{d}z + \left(z + \frac{xy}{w}\right)\mathrm{d}y\right\}^2 + \textit{cyclic terms}.$$

Multiplying out, collecting like terms and inspecting the coefficients of $\mathrm{d}x^2$, $\mathrm{d}x\mathrm{d}y$ etc., we see that

$$g_{ij} = \frac{4}{w^2}\begin{pmatrix} w^2 + x^2 & xy & xz \\ xy & w^2 + y^2 & yz \\ xz & yz & w^2 + z^2 \end{pmatrix}$$

and we can calculate $|g| = 1/w^2$. Inverting the metric, we get

$$g^{ij} = \frac{1}{4}\begin{pmatrix}(1-x^2) & -xy & -xz\\ -xy & (1-y^2) & -yz\\ -xz & -yz & (1-z^2)\end{pmatrix}.$$

Now, we want to switch to using w as a coordinate, and to find expressions for Δf where $f(w)$ is a function only of w. To this end, we have

$$\nabla f = \frac{\partial f}{\partial w}\mathrm{d}w = -\frac{1}{w}\frac{\partial f}{\partial w}\left(x\mathrm{d}x + y\mathrm{d}y + z\mathrm{d}z\right),$$
$$g^{ij}(\nabla_f)_j = -\frac{1}{4w}\frac{\partial f}{\partial w}\begin{pmatrix}(1-x^2) & -xy & -xz\\ -xy & (1-y^2) & -yz\\ -xz & -yz & (1-z^2)\end{pmatrix}\begin{pmatrix}x\\ y\\ z\end{pmatrix} = -\frac{w}{4}\frac{\partial f}{\partial w}\begin{pmatrix}x\\ y\\ z\end{pmatrix},$$

so

$$\Delta f = w\,\partial_i\left(\frac{1}{w}g^{ij}(\nabla f)_j\right) = -\frac{3w}{4}\frac{\partial f}{\partial w} + \frac{1-w^2}{4}\frac{\partial^2 f}{\partial w^2}.$$

If we make the substitution $w = \cos(\alpha/2)$, where α is the angle of the corresponding rotation, then $\mathrm{d}w = -\frac{1}{2}\sin(\alpha/2)\mathrm{d}\alpha$, and we get

$$\Delta f = \cot(\alpha/2)\frac{\partial f}{\partial \alpha} + \frac{\partial^2 f}{\partial \alpha^2}.$$

To find the transition probabilities, we must solve the Fokker–Planck equation

$$\frac{\partial q}{\partial t} = \frac{1}{2}\Delta q$$

on $SO(3)$, subject to the initial condition of a delta mass at I. By symmetry, we know the solution will be rotationally symmetric, so we can write the solution as $q(\alpha, t)$. Now, we look for separable solutions of the form $q(\alpha, t) = T(t)A(\alpha)$. We see that we must have

$$\frac{1}{T}\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{1}{2A}\left(\cot(\alpha/2)\frac{\mathrm{d}A}{\mathrm{d}\alpha} + \frac{\mathrm{d}^2 A}{\mathrm{d}\alpha^2}\right).$$

Separating the two equations, we see that we require

$$\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{1}{2}\lambda T, \qquad \cot(\alpha/2)\frac{\mathrm{d}A}{\mathrm{d}\alpha} + \frac{\mathrm{d}^2 A}{\mathrm{d}\alpha^2} = \lambda A,$$

for some fixed λ. The first equation has solution $T(t) = e^{\lambda t/2}$, while a solution to the second is given by

$$A(\alpha) = \frac{\sin\left(\left(\mu + \frac{1}{2}\right)\alpha\right)}{\sin(\alpha/2)},$$

where μ satisfies $-\mu(\mu+1) = \lambda$. In addition, the boundary conditions force μ to be an integer. Combining these expressions, we see that the solution is of the form

$$q(\alpha, t) = \sum_{\ell=0}^{\infty}\beta_\ell e^{-\ell(\ell+1)t/2}\frac{\sin\left(\left(\ell + \frac{1}{2}\right)\alpha\right)}{\sin(\alpha/2)}$$

for some coefficients β_ℓ. Finally, we have the initial condition that $q(\alpha, 0) = 0$ for $\alpha > 0$ and $\int_{SO(3)} q(\mathbf{x}, 0) f(\mathbf{x}) \mathrm{d}\nu(\mathbf{x}) = f(I)$ where ν is the uniform probability measure on $SO(3)$. Up to a scaling factor, this is satisfied if and only if $\beta_\ell \propto (2\ell + 1)$. Putting this all together, we obtain Equation (29).

J.5. Mixture of wrapped normal distributions on $SO(3)$

We consider modelling a mixture of wrapped normal distributions on $SO(3)$. The wrapped normal distribution $\mathcal{N}^W(\mathbf{x} \mid \mu, \sigma^2)$ with mean μ and variance σ^2 is defined here as the transformed distribution via sampling $\mathbf{w} \sim \mathcal{N}(\mathbf{w} \mid 0, \sigma^2)$, where $\mathbf{w} \in \mathbb{R}^{3\times 3}$, from the standard normal distribution with variance σ^2, projecting $\mathbf{w}$ onto the tangent space via $\mathbf{v} = \frac{\mathbf{w} - \mathbf{w}^T}{2}$, then applying the exponential map $\mathbf{x} = \exp_\mu(\mathbf{v})$ at μ. While we could apply standard parametric learning methods which involve learning of $\{\mu_m, \sigma_m\}$ directly, we do not rely on the specific form of the data distribution p_{data}, which allows us to model different distributions flexibly. We consider modelling of a mixture of wrapped normal distributions with $M = 16$ mixtures.

We apply a time-rescaling for the noising process, which is given by $\mathcal{L} = \partial_t + \frac{1}{2}\beta(t)\Delta$ with the linear β schedule given in Equation (28). Then, the reverse process is generated by $\mathcal{K} = \partial_t + \beta(t) s_\theta(\mathbf{x}, t) \cdot \nabla + \frac{1}{2}\beta(t)\Delta$. We use an MLP with 5 layers and 512 hidden units in each layer to output a vector of dimension 3 parameterising $\{s^i_\theta(\mathbf{x}, t)\}^3_{i=1}$. We train the network using the Adam optimiser with batch size 512 and learning rate 0.0002 with a cosine annealing schedule for 100000 iterations.

We learn both the unconditional distribution $p_{\text{data}}(\mathbf{x})$ and the conditional distribution $p_{\text{data}}(\mathbf{x}|m)$ when conditioned on the cluster member m. In the conditional case, we learn a conditional score model $s_\theta(\mathbf{x}, m, t)$ under the same settings.

Fig. 10 shows the results from our conditional model for $p_{\text{data}}(\mathbf{x}|m)$, where we compare the unwrapped distributions in the tangent space between the ground truth normal distribution and the modelled distribution of mixture member $m = 1$, and plot a representative sample from our conditional model. We see that our model targets the correct mixture accurately. Our visualisations of distributions on $SO(3)$ are adapted from Murphy et al. (2021).

We compare our method to the method of De Bortoli et al. (2022), in which the denoising diffusion model for this task is trained by simulating the forward process using the Geodesic Random Walk and using the DSM loss with Varadhan's approximation, rather than using the analytic transition densities given in Appendix J.4.1 as we do. We compare the two methods using the learned models' test-set log-likelihood, calculated using the probability flow ODE as in De Bortoli et al. (2022), as well as the average time per training iteration. Our results are shown in Table 2. We see that both methods achieve comparable log-likelihoods, but our method is about 15% more efficient during training since having the analytic transition densities means that we can simulate the forward noising process in a single step.

J.6. Pose estimation on the SYMSOL dataset

We give details for the pose estimation task on the SYMSOL dataset. We use a similar network design for the conditional score $s_\theta(\mathbf{x}_t, \boldsymbol{\xi}, t)$ as Murphy et al. (2021), composed of

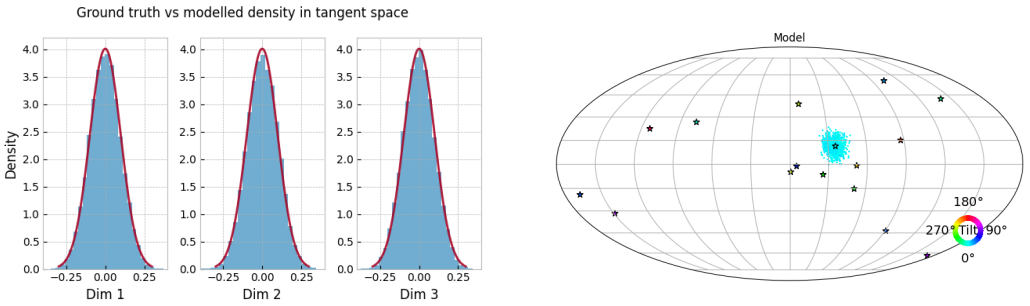

Fig. 10. (Left) Histogram of samples from our model conditioned on the mixture member $m = 1$ compared to the ground truth normal density, represented in the tangent space of $SO(3)$. (Right) Conditional samples from the model for $m = 1$. The axis of rotation and rotation angle are represented by position and colour respectively.

Table 2. Test set log-likelihood and time per training iteration for denoising models on $SO(3)$. Mean and standard deviation reported over 5 seeds.

	$M = 16$	$M = 32$	$M = 64$	Time per iteration (ms)
De Bortoli et al. (2022)	0.864±0.026	0.174±0.025	-0.516±0.016	55.18±2.783
Analytic (ours)	0.872±0.026	0.175±0.025	-0.515±0.016	47.23±2.134

a vision recognition model for processing the input images $\boldsymbol{\xi}$, and an MLP for outputting the score. For the vision recognition model, we utilise pretrained ResNet-50 backbone without the final fully-connected classification layer, which outputs a 2048-dimensional embedding. We next get sinusoidal positional embeddings of $\mathbf{x}_t$ and t, use linear layers to transform all embeddings into 256 dimensions and take the summed embedding. This also allows efficient computations of embeddings with a single $\boldsymbol{\xi}$ and multiple values of $(\mathbf{x}_t, t)$ as the computationally expensive forward pass through the vision recognition model only needs to be taken once. Thus, we simulate a small number of $(\mathbf{x}_t, t)$ pairs given each pair $(\mathbf{x}_0, \boldsymbol{\xi})$ at each step for more efficient training. We finally pass the embedding into an MLP with 3 layers and 256 hidden units in each layer.

Compared to the Implicit-PDF methodology by Murphy et al. (2021), which maintains a grid on $SO(3)$ and approximates the density pointwise, our DMM model directly learns a sampling method and does not require maintaining a grid. Therefore, our method is more general and not specific to $SO(3)$. For our implementation, we modify their network structure to take in the time t, and output the score parameterisation of dimension 3 as opposed to the unnormalised log density of dimension 1. We optimise the network using the Adam optimiser with batch size 128 and learning rate 0.0001 with a cosine annealing schedule for 100000 iterations.

We include further visualisations of the generated samples when conditioned on 2D views of different shapes in Fig. 11. As shown in the plots, the samples generated using DMM are all close to the ground truth and cover all modes of the class of rotational symmetries.

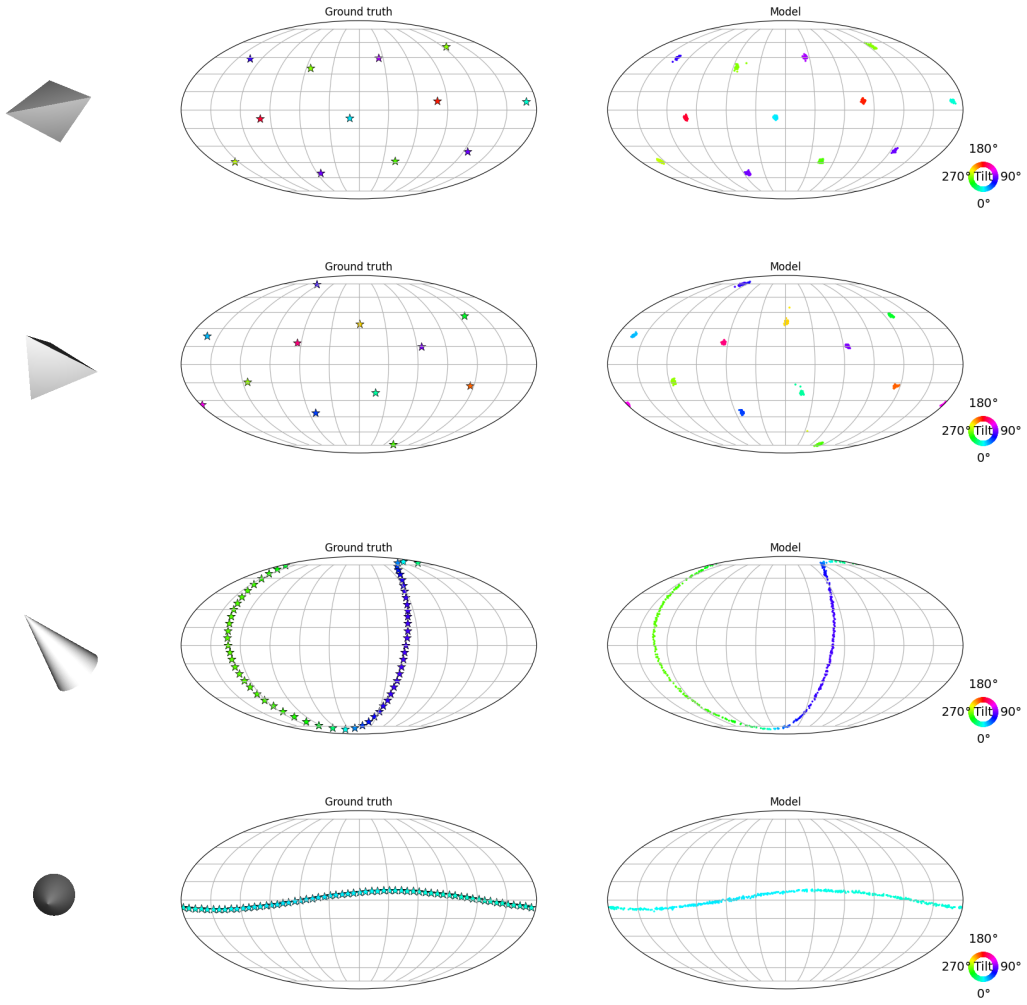

Fig. 11. Samples from the ground truth (plotted as stars, middle) and our pose estimation DMM (right) conditioned on 2D views of shapes (left). The axis of rotation and rotation angle are represented by position and colour respectively.

Table 3. True test data log-likelihood compared to the DMM model ELBO as given by (8) using the ISM loss for the mixture of Dirichlet example. Mean and standard deviation reported over 5 seeds.

Dimension of simplex	$N = 3$	$N = 5$	$N = 10$	$N = 20$
Data	1.321±0.340	4.122±0.242	15.288±0.389	45.914±0.694
Model	1.158±0.160	4.017±0.208	15.061±0.428	45.494±0.698

J.7. Approximation of distributions over measures using Wright–Fisher diffusions

Finally, we evaluate the Wright–Fisher diffusion framework from Appendix F.4 for modeling distributions over measures on a finite state space. We test our framework by attempting to model mixtures of Dirichlet distributions $p_{\text{data}}(\mathbf{x}) = \frac{1}{M}\sum_{m=1}^{M} \text{Dirichlet}(\alpha_m)$ with parameters $\alpha_m \in \mathbb{R}^N$. We consider $M = 4$ mixtures and vary the number of dimensions N of the simplex.

As in Appendix F.4, we use a Wright–Fisher diffusion with $q_{ij} = \vartheta_j$ for all $i \neq j$ as our noising process, and set $\vartheta_j = 3$ for all $j = 1, \dots, N$. We also apply a time rescaling to the forward process as in Equation (28). We set $\beta_{\min} = 0.001$ and $\beta_{\max}$ is selected using a grid search from $0.5, 1, 2$. We simulate the forward diffusion process using the exact simulation algorithm of Jenkins and Spanò (2017), which exploits the eigenfunction decomposition of the Wright–Fisher process transition function given in Equation (21) and works by sampling from the ancestral process $A^{\Theta}_{\infty}(t)$ whose distribution is determined by the functions $\{d^{\Theta}_n(t) : n = 0, 1, \dots\}$. For very small times t, we also use a normal approximation for simulating $A^{\Theta}_{\infty}(t)$. For more details, we refer the reader to Jenkins and Spanò (2017).

We learn the score network with the parameterisation $s^i_\theta(\mathbf{p}, t) = p_i \partial(\log \beta(\mathbf{p}, t))/\partial p_i$ using the implicit score matching loss (27). We parameterise $s_\theta(\mathbf{p}, t)$ using an MLP with 4 layers and 512 hidden units in each layer to output a vector of dimension N. We train the network using the Adam optimiser with batch size 128 and learning rate 0.0001 with a cosine annealing schedule for 100000 iterations.

We visualise the results of this experiment in Fig. 7 for a 3-dimensional example. As can be seen, the DMM model is able to learn the ground truth distribution very accurately. We also report in Table 3 the ground truth log-likelihood of the data distribution $p_{\text{data}}(\mathbf{x})$ and the ELBO of the DMM model given by (8) using the ISM loss, as the number of dimensions N increases. We observe that the model's ELBO is consistently close to the true data log-likelihood, which demonstrates the scalability of the DMM model.

References

Cattiaux, P., G. Conforti, I. Gentil, and C. Léonard (2021). Time Reversal of Diffusion Processes under a Finite Entropy Condition. *arXiv:2104.07708*.

Dong, R. (2003). Feller Processes and Semigroups. *Lecture notes, UC Berkeley, https://www.stat.berkeley.edu/~pitman/s205s03/lecture27.pdf*.

Durkan, C., A. Bekasov, I. Murray, and G. Papamakarios (2019). Neural Spline Flows. *NeurIPS*.

Durkan, C., I. Murray, and G. Papamakarios (2020). On Contrastive Learning for Likelihood-free Inference. *ICML*.

Ethier, S. N. and T. G. Kurtz (1993). Fleming–Viot Processes in Population Genetics. *SIAM Journal on Control and Optimization 31*, 345–386.

Fearnhead, P. and D. Prangle (2012). Constructing Summary Statistics for Approximate Bayesian Computation: Semi-automatic Approximate Bayesian Computation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 74*(3), 419–474.

Greenberg, D. S., M. Nonnenmacher, and J. H. Macke (2019). Automatic Posterior Transformation for Likelihood-Free Inference. *ICML*.

Jenkins, P. A. and D. Spanò (2017). Exact Simulation of the Wright–Fisher Diffusion. *The Annals of Applied Probability 27*(3).

Karatzas, I. and S. E. Shreve (1991). *Brownian Motion and Stochastic Calculus*. Springer Science & Business Media.

Leach, A., S. M. Schmon, M. T. Degiacomi, and C. G. Willcocks (2022). Denoising Diffusion Probabilistic Models on SO(3) for Rotational Alignment. *ICLR 2022 Workshop on Geometrical and Topological Representation Learning*.

LeCun, Y., C. Cortes, and C. Burges (2010). MNIST handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*.

Molchanov, S. A. (1968). Strong Feller Property of Diffusion Processes on Smooth Manifolds. *Theory of Probability & Its Applications 13*, 471–475.

Métivier, M. (1982). *Semimartingales*. De Gruyter.

Palmowski, Z. and T. Rolski (2002). A Technique for Exponential Change of Measure for Markov Processes. *Bernoulli 8*, 767–785.

Papamakarios, G., D. C. Sterratt, and I. Murray (2019). Sequential Neural Likelihood: Fast Likelihood-free Inference with Autoregressive Flows. *AISTATS*.

Prangle, D. (2020). gk: An R Package for the g-and-k and Generalised g-and-h Distributions. *The R Journal 12*(1), 7–20.

Pulido, S. (2011). Semimartingales and stochastic integration. *Lecture Notes, CMU, https://www.andrew.cmu.edu/user/calmost/pdfs/21-882-int_lec.pdf*.

Russakovsky, O., J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision 115*(3), 211–252.

Schilling, R. L. and L. Partzsch (2012). *Brownian Motion: An Introduction to Stochastic Processes*. De Gruyter.

Sohl-Dickstein, J., P. B. Battaglino, and M. R. Deweese (2011). New Method for Parameter Estimation in Probabilistic Models: Minimum Probability Flow. *Physical Review Letters 107*.

Taylor, M. E. (2011). *Partial Differential Equations I: Basic Theory*. Springer.

Tejero-Cantero, A., J. Boelts, M. Deistler, J.-M. Lueckmann, C. Durkan, P. J. Gonçalves, D. S. Greenberg, and J. H. Macke (2020). sbi: A Toolkit for Simulation-based Inference. *Journal of Open Source Software 5*(52), 2505.

Vincent, P., H. Larochelle, Y. Bengio, and P. A. Manzagol (2008). Extracting and Composing Robust Features with Denoising Autoencoders. *ICML*.

Yosida, K. (1965). *Functional Analysis*. Springer Science & Business Media.