Score Matching and Flows

Luca Raffo
EPFL, Institute of Mathematics - luca.raffo@epfl.ch
Capital Fund Management - luca.raffo@cfm.com

June 2025

Contents

1 Introduction

Score matching, introduced in [3], yields a new heuristic to estimate continuous statistical models where the probability density function is known only up to a multiplicative normalization constant. The method is shown to be locally consistent under identifiability of the model, and the estimation does not require to compute the normalization constant.

Here we propose a new point of view that shows that score matching is equivalent to searching for the model that minimizes the Wasserstein gradient (under empirical expectation) of the KL divergence between the real density and the estimated one.

This framework can be generalized to different functionals and may lead to new methods for parametric statistical inference.

Then we move on to show a deep connection between score matching and minimum probability flow. The latter, introduced in [4], is a different method for estimating statistical models, initially developed for discrete domains such as Ising models, but easily adapted to continuous problems.

2 Background on Score Matching

As in usual statistical inference frameworks, we start from a given set of datapoints $D = \{x^{(1)}, ..., x^{(n)}\}$ with $x^{(i)} \in \mathbb{R}^d$, sampled via $X^{(1)}, ..., X^{(n)} \overset{\text{iid}}{\sim} \mu_g$, where μ_g is the real (unknown) distribution whose density with respect to the Lebesgue measure is g.
We want to model the distribution with π_θ, absolutely continuous with respect to Lebesgue, and with density $f_\theta(x) = \frac{1}{Z}e^{-H_\theta(x)}$.

Score matching reads

$$\begin{aligned}\theta_{SM} &= \arg\min_\theta \mathbb{E}_g[\|\nabla_x \log f_\theta - \nabla_x \log g\|_2^2] \\ &= \arg\min_\theta \mathbb{E}_g[(\nabla_x H_\theta)^2 - 2\Delta_x H_\theta] \text{ after integrating by parts},\end{aligned}$$

whose sample version is

$$\begin{aligned}\theta^*_{SM} &= \arg\min_\theta \mathbb{E}_D[\|\nabla_x \log f_\theta - \nabla_x \log g\|_2^2] \\ &= \arg\min_\theta \mathbb{E}_D[(\nabla_x H_\theta)^2 - 2\Delta_x H_\theta],\end{aligned}$$

which does not require the normalization constant Z and can be computed from the data.

In [3] it is shown that θ_{SM} is locally consistent if the model is identifiable (i.e. under the condition that if $\theta_1 \neq \theta_2$ then $\pi_{\theta_1} \neq \pi_{\theta_2}$), and the sample version is asymptotically equivalent to the population one due to the strong law of large numbers.

3 Background on Wasserstein Gradient Flows

The main objective of this section is to unify the notation regarding flows of measures and to define properly Wasserstein gradient flows.

3.1 Flows of measures

Let us sample $X_0 \sim \mu_0$, with $d\mu_0 = f_0 \, d\lambda$ and let $v_t : \mathbb{R}^d \to \mathbb{R}$ be any vector field. If we evolve our particle via

$$\dot{X}_t = v_t(X_t), \tag{1}$$

we find out that $\mu_t := Law(X_t)$ is absolutely continuous with respect to Lebesgue, it has finite second moment, and its density satisfies the continuity equation, i.e. it satisfies

$$\partial_t f_t + \nabla \cdot (v_t f_t) = 0 \tag{2}$$

in weak sense. The proof can be found in [1].

3.2 Background on Wasserstein spaces

Consult manuscript I.

3.3 Wasserstein gradient flows

Given a functional $\mathcal{F} : \mathcal{P}_2^{ac}(\lambda) \to \mathbb{R}$ with bounded first variation, we define its Wasserstein gradient at $\mu \in \mathcal{P}_2^{ac}(\lambda)$ as

$$\begin{aligned} \nabla_{\mathcal{W}_2}\mathcal{F}(\mu) : \mathbb{R}^d &\to \mathbb{R}^d \\ x &\mapsto \nabla[\delta\mathcal{F}(\mu)](x). \end{aligned}$$

Now we fix a functional $\mathcal{F}$ with bounded first variation, and use $-\nabla_{\mathcal{W}_2}\mathcal{F}(\mu_t)$ as our vector field v_t in (1), so that μ_t will evolve via

$$\partial_t f_t + \nabla \cdot (-\nabla_{\mathcal{W}_2}\mathcal{F}(\mu_t) f_t) = 0, \tag{3}$$

that is known as *Wasserstein gradient flow* of μ_t with respect to $\mathcal{F}$, started at μ_0.

There are many vector fields v_t such that the ODE (1) induces the PDE (3), it turns out that the *most economical* one, i.e. the one which minimizes $\|v_t\|^2_{L_2(\mu_t)}$ is

$$v_t = -\nabla_{\mathcal{W}_2}\mathcal{F}(\mu_t). \tag{4}$$

Again, we suggest consulting [1] for a proof.

Clearly if a functional $\mathcal{F}$ is displacement convex, then it has a unique minima μ^*.

The main result for this section, denoted in [2] as Poljak–Łojasiewicz inequality, states that the Wasserstein gradient flow with respect to a (strongly) displacement convex $\mathcal{F}$, started at any $\mu_0 \in \mathcal{P}_2^{ac}(\lambda)$, converges exponentially fast towards the unique minimizer $\mu^* \in \mathcal{P}_2^{ac}(\lambda)$.

It turns out that $\mathcal{F}(\cdot) = \mathcal{D}_{KL}(\cdot\|\pi_\theta)$ is (strongly) displacement convex, so it is reasonable trying to minimize it via Wasserstein gradient flows.

4 Score Matching and Wasserstein Gradient Flows

Let us come back to our usual framework of μ_g and π_θ. The idea is that if $\mu_g \approx \pi_\theta$, then

$$\mathcal{D}_{KL}(\mu_g\|\pi_\theta) \approx 0, \tag{5}$$

so that the Wasserstein gradient flow of $\mathcal{D}_{KL}(\cdot\|\pi_\theta)$ (which is playing the role of $\mathcal{F}(\cdot)$) started from $\mu_0 := \mu_g$ will be almost stationary. For a sample $X_0 \sim \mu_g$, since $\dot{X}_0 = -\nabla_{\mathcal{W}_2}\mathcal{D}_{KL}(\mu_g||\pi_\theta)(X_0)$ is the starting point of the *most economical* ODE inducing (3), the requirement (5) naturally translates into finding

$$\theta^* := \arg\min_\theta \mathbb{E}_g[\| - \nabla_{\mathcal{W}_2}\mathcal{D}_{KL}(\mu_g\|\pi_\theta)\|_2^2]. \tag{6}$$

But we know that $\nabla_{\mathcal{W}_2}\mathcal{D}_{KL}(\cdot\|\pi_\theta) = \nabla_x H_\theta + \nabla_x \log g$ (consult manuscript I), so that (6) is aiming to find

$$\begin{aligned}\theta^* &= \arg\min_\theta \mathbb{E}_g[(-\nabla_x H_\theta - \nabla_x \log g)^2] \\ &= \arg\min_\theta \mathbb{E}_g[(\nabla_x \log f_\theta - \nabla_x \log g)^2]\end{aligned}$$

that is precisely the starting point of score matching.

5 Score Matching and Minimum Probability Flow

Minimum probability flow, introduced in [4], is another method for statistical inference. In the continuous framework, in can be summarized as

$$\theta_{MPF} = \arg\min_\theta \mathbb{E}_g[|\frac{d}{dt}\mathcal{D}_{KL}(\mu_g\|\mu_t)|_{t=0}|],$$

with μ_t following the Fokker-Planck[1] equation with potential π_θ.

It has been shown in [4], using brute force computations, that the function that minimum probability flow is trying to minimize is the same as score matching; in this section we want to show it from another perspective.

The main idea is that the Wasserstein gradient flow of $\mathcal{D}_{KL}(\cdot\|\pi_\theta)$ is the Fokker–Planck equation with potential H_θ, i.e.

$$\partial_t f_t + \nabla\cdot(-\nabla(\log f_t + H_\theta) f_t) = 0 \quad\Longleftrightarrow\quad \partial_t f_t = \Delta f_t + \nabla\cdot(f_t\nabla H_\theta),$$

and we can show that score matching is in turn equivalent to minimizing $\mathbb{E}_g[|\frac{d}{dt}\mathcal{D}_{KL}(\mu_t\|\pi_\theta)|_{t=0}|]$, as μ_t satisfies the Fokker Planck equation with potential H_θ (IT IS STILL NOT TRIVIAL TO SHOW FORMALLY THAT THEY ARE EQUIVALENT.)

6 Langevin Matching

Motivated by the previous sections, we propose a new algorithm for parameter inference in continuous models, inspired by the idea of matching the temporal evolution of probability distributions under Langevin dynamics.

[1] in practice we use a discretized version.

6.1 Extending score matching

We begin by recalling the classical score matching algorithm (from now on denoted as SM):

1. Given data samples $x_1, \dots, x_n \sim \mu_g$.
2. Model unknown μ_g with unnormalized density $\pi_\theta(x) \propto \exp(-H_\theta(x))$.
3. Define the objective function (the expectation over g has to be interpreted as an empirical expectation):

   $$\mathcal{L}^{(1)}_{\mathrm{SM}}(\theta) := \mathbb{E}_g \left[\|\nabla_x H_\theta(x)\|_2^2 - 2\Delta_x H_\theta(x) \right],$$

   which is equivalent to

   $$\mathcal{L}^{(2)}_{\mathrm{SM}}(\theta) := \mathbb{E}_g[\|\nabla_{\mathcal{W}_2} \mathcal{D}_{KL}(\mu_g \| \pi_\theta)\|_2^2],$$

   which again is equivalent to

   $$\mathcal{L}^{(3)}_{\mathrm{SM}}(\theta) := \mathbb{E}_g[|\frac{d}{dt}\mathcal{D}_{KL}(\mu_t \| \pi_\theta)|_{t=0}|],$$

   where μ_t is the solution of the Fokker–Planck equation:

   $$\partial_t \mu_t = \nabla \cdot (\mu_t \nabla H_\theta) + \Delta \mu_t, \text{ with } \mu_0 = \mu_g.$$

4. Optimize over θ with any convex optimization algorithm. We show below an instance with gradient descent.

Algorithm 1 SM

```
Initialize parameters $\theta_0$
for $k = 0$ to $K - 1$ do
    Compute gradient $\nabla_\theta \mathcal{L}_{\mathrm{SM}}(\theta_k)$
    Update: $\theta_{k+1} \leftarrow \theta_k - \eta \nabla_\theta \mathcal{L}_{\mathrm{SM}}(\theta_k)$
end for
Return $\theta^*$ as estimate of optimal parameter $\theta_{SM}$
```

We show below a graphical representation of the algorithm.

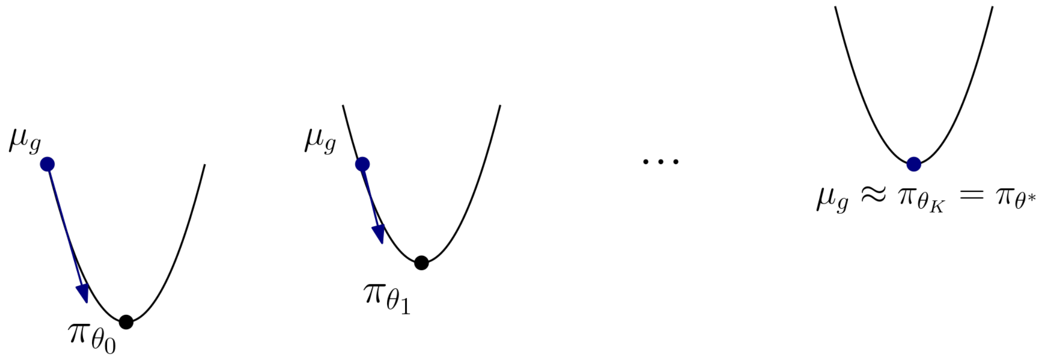

Figure 1: SM evolution.

The black line represent the Wasserstein gradient flow of the KL divergence, a.k.a. the Fokker-Planck evolution with potential H_θ, and the blue arrow represent the Wasserstein gradient.

Notice that while the evolution regards μ_g moving towards π_θ, in the algorithm μ_g is fixed, and the optimization takes place through θ.

In the implementation we substitute μ_g with the empirical measure.

Motivated by this understanding, instead of trying to minimize iteratively the Wasserstein gradient, our idea is to evolve at each step our empirical measure μ_g along the gradient flow (we know that Langevin diffusion with potential H_θ evolves the measure via Fokker-Planck with the same potential), and then optimize this in θ with numerical differentiation.

The algorithm reads as follows.

1. Given data samples $x_1, \dots, x_n \sim \mu_g$.
2. Model unknown μ_g with unnormalized density $\pi_\theta(x) \propto \exp(-H_\theta(x))$.
3. Define the objective function (the expectation over g has to be interpreted as an empirical expectation):

$$\mathcal{L}_{LM}(\theta) := \mathbb{E}_g[\mathcal{D}(\mu_g, \pi_\theta)],$$

for some discrepancy $\mathcal{D} : \mathcal{P} \times \mathcal{P} \to \mathbb{R}$.

4. Optimize over θ with any convex optimization algorithm. We show below an instance with gradient descent.

Algorithm 2 LM

```
Initialize parameters $\theta_0$
for $k = 0$ to $K - 1$ do
    Evolve $\mu_g$ with $n$ steps of Langevin with potential $H_{\theta_k}$
    Compute gradient $\nabla_\theta \mathcal{L}_{LM}(\theta_k)$
    Update: $\theta_{k+1} \leftarrow \theta_k - \eta \nabla_\theta \mathcal{L}_{LM}(\theta_k)$
end for
Return $\theta^*$ as estimate of optimal parameter $\theta_{LM}$
```

We show below a graphical representation of the algorithm.

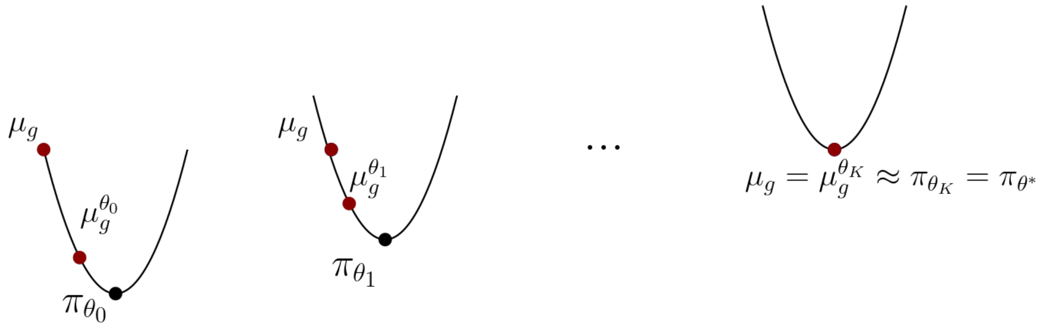

Figure 2: LM evolution.

In practice we have a variety of possibilities for $\mathcal{D}$, such as *Sinkhorn, MMD, Sliced Wasserstein* (a brief introduction to MMD can be found in manuscript II, Sinkhorn and Sliced Wasserstein are approximations of the Wasserstein distance) and others.

Notice that if $\mathcal{D}(\mu_g, \pi_{\theta_k}) = \frac{1}{\varepsilon}\left[\mathcal{D}_{KL}(\mu_0 \| \pi_{\theta_k}) - \mathcal{D}_{KL}(\mu_\varepsilon \| \pi_{\theta_k})\right]$ as $\epsilon \to 0$ we get back SM.

From now on we will denoted the algorithms with LM *name of the loss*, or LM to stay more generic.

6.2 Numerical experiments

We have tried to implement LM for different choices of the loss function, using SM as a benchmark. The comparison has been carried out for inferring parameters of gaussian distributions in dimensions $1, 3, 5, 10$ and 50 using a variety of loss functions. The optimization has been carried out both with Adam and SGD yielding same results. Using an adaptive learning rate does not seem to improve the inference.

We anticipate that SM has better performances both in terms of inference and computational efficiency. The latter is trivially due to the fact that numerical differentiation is expensive for LM, with any loss. The inferior inference performance of LM may be attributed to several factors, including the approximation errors introduced by discretized Langevin dynamics and the additional noise from stochastic updates.

We show below the comparison of reconstruction errors after convergence, for a number of accessible samples fixed at 2000. For SM, the optimization has been carried out with Adam. For LM, we used Adam up to dimension 10 to get more stable results, but we relied on SGD for the last case due to high non-convexity of the loss landascape.

Dim	**SM**	**LM Moments**	**LM Sinkhorn**	**LM MMD (Gauss $\sigma^2 = 1$)**	**LM Energy**	**LM SlicedWass**
1	0.022, 0.007	0.054, 0.019	0.065, 0.013	0.032, 0.062	0.046, 0.060	0.052, 0.022
3	0.038, 0.050	0.051, 0.055	0.035, 0.070	0.086, 0.054	0.062, 0.046	0.051, 0.053
5	0.015, 0.042	0.079, 0.083	local minima	0.081, 0.116	0.077, 0.093	0.076, 0.086
10	0.021, 0.055	0.040, 0.098	local minima	0.044, 0.176	0.044, 0.176	0.040, 0.097
50	0.025, 0.126	0.059, 0.216	local minima	σ too small	local minima	local minima

Table 1: Parameters reconstruction errors for each method and dimension. The first entry regards the mean reconstruction error (measured with square norm), the second one the covariance reconstruction error (measured with Frobenius norm).

Dim	SM	LM Moments	LM Sinkhorn	LM MMD (Gauss $\sigma^2 = 1$)	LM Energy	LM SlicedWass
1	0.0040	0.1609	0.4690	0.2001	0.1931	0.1974
3	0.0046	0.1623	0.5110	0.2075	0.1976	0.1968
5	0.0036	0.1677	0.7439	0.2094	0.1947	0.2468
10	0.0035	0.1830	0.5276	0.2591	0.2549	0.2942
50	0.0036	0.2815	0.5477	0.3174	0.3264	0.3523

Table 2: Execution time (in seconds) for each method and dimension, over the number of iterations.

We show below the evolution of the losses trajectories against the number of samples upon which they are trained. We plot SM and LM Moments only in dimension 3 for the sake of brevity, but they all show similar behavior.

Remarkably, despite slightly worse inferences and a slower time of execution, LM Moments converges much faster than SM.

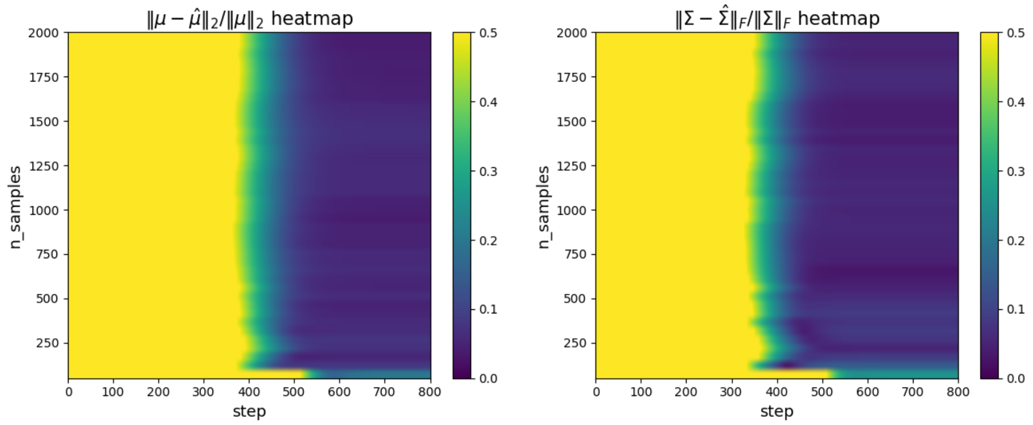

Figure 3: SM trajectories losses against number of samples in dimension 3.

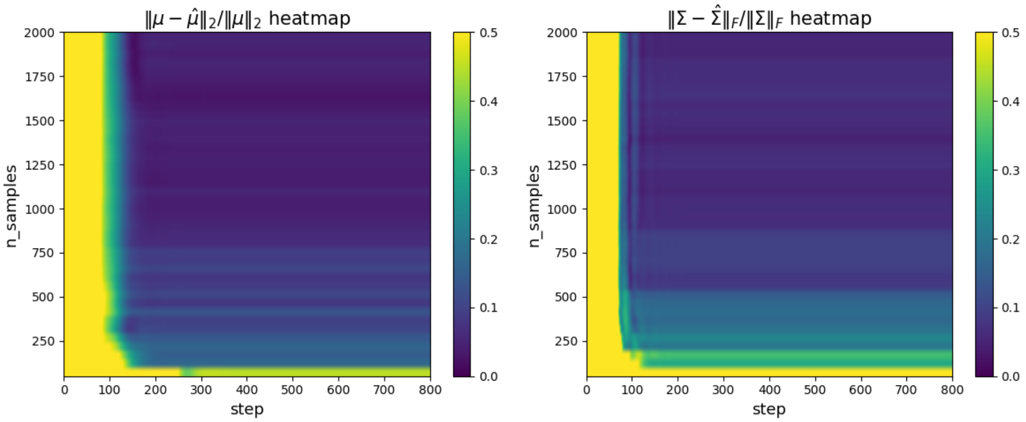

Figure 4: LM Moments trajectories losses against number of samples in dimension 3.

7 Augmented SM

We propose further a new algorithm, based on SM, which relies on the previous arguments to augment data during training time.

It simply reads as follows:

1. Given data samples $x_1, \dots, x_n \sim \mu_g$.
2. Model unknown μ_g with unnormalized density $\pi_\theta(x) \propto \exp(-H_\theta(x))$.
3. Define the objective function (the expectation over g has to be interpreted as an empirical expectation):

$$\mathcal{L}^{(1)}_{\mathrm{SM}}(\theta) := \mathbb{E}_g\left[\|\nabla_x H_\theta(x)\|_2^2 - 2\Delta_x H_\theta(x)\right].$$

4. Do the following algorithm.

Algorithm 3 Augmented SM

```
Initialize parameters $\theta_0$. Fix $\epsilon > 0$.
for $k = 0$ to $K - 1$ do
    Compute gradient $\nabla_\theta \mathcal{L}_{\text{SM}}(\theta_k)$
    Update: $\theta_{k+1} \leftarrow \theta_k - \eta \nabla_\theta \mathcal{L}_{\text{SM}}(\theta_k)$
    if $\|\theta_{k+1} - \theta_k\| < \epsilon$ then
        Duplicate $\mu_g$, call the new data $\mu'_g$
        Evolve $\mu'_g$ with $n$ steps of Langevin with potential $H_{\theta_k}$
        Add the evolved data to the original batch
    end if
end for
Return $\theta^*$ as estimate of optimal parameter $\theta_{SM}$
```

We plot below the evolution of the losses trajectories against the number of samples. As expected, we see a tradeoff: when the number of samples is low, introducing synthetic data improves the inference and the time of convergence, but as the number of samples becomes bigger we see the performances get worse since we are introducing too much bias.

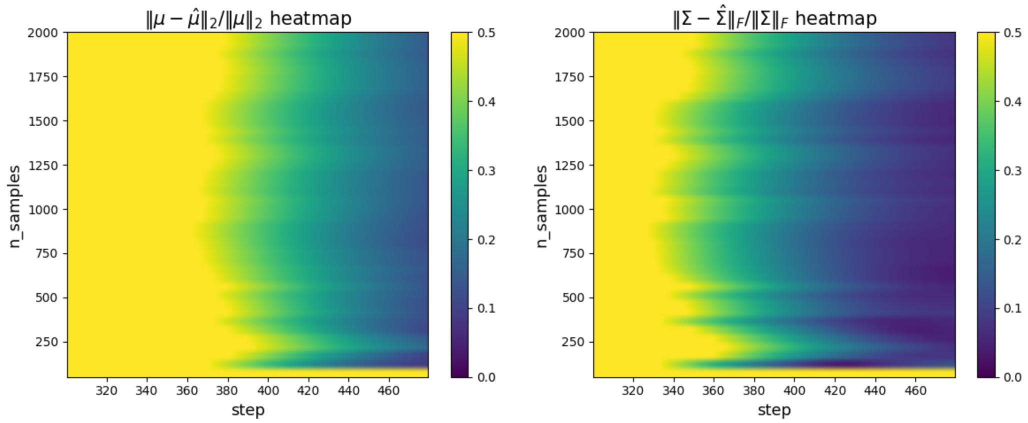

Figure 5: SM trajectories losses against number of samples in dimension 3.

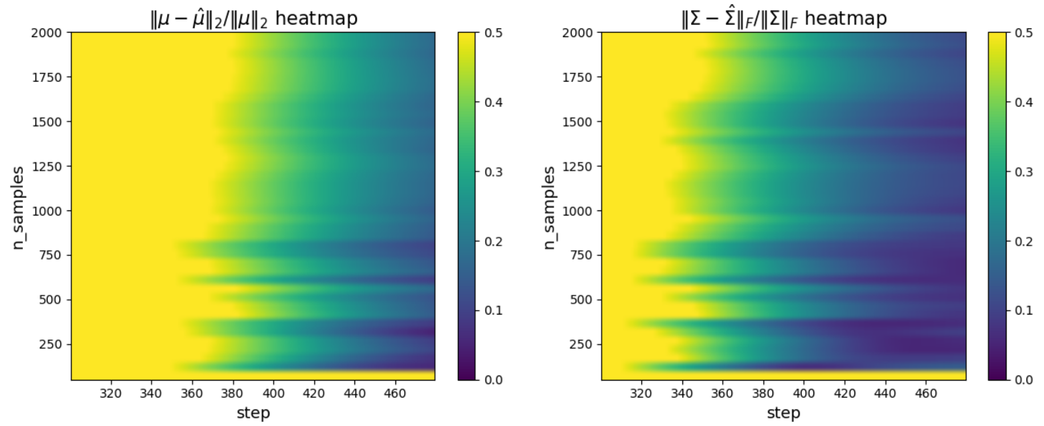

Figure 6: Augmented SM trajectories losses against number of samples in dimension 3.

References

[1] L. Ambrosio, N. Gigli, and G. Savare. *Gradient Flows: In Metric Spaces and in the Space of Probability Measures*. Lectures in Mathematics. ETH Zürich. Birkhäuser Basel, 2008.

[2] Sinho Chewi, Jonathan Niles-Weed, and Philippe Rigollet. Statistical optimal transport, 2024.

[3] Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005.

[4] Jascha Sohl-Dickstein, Peter Battaglino, and Michael Robert DeWeese. A new method for parameter estimation in probabilistic models: Minimum probability flow. *CoRR*, abs/2007.09240, 2020.