Project on Sliced-Wasserstein Flows: Nonparametric Generative Modeling via Optimal Transport and Diffusions

Hippolyte PILCHEN (MVA ENS Paris-Saclay)

Computational Optimal Transport project – 01/17/2024

Abstract

The paper of interest [1] presents a non parametric implicit generative model. It enables to generate samples which match the underlying distribution of previous observations. While researchs on generative modeling are currently very popular a lot of generative models do not have theoretical guarantees. The authors designed an algorithm to solve a precise optimization problem and derived non-asymptotic error guarantees on their solution. The model approaches the targeted distribution, which is theoretically proved, but also enables to generate new diversified samples (avoiding over-fitting on training data) which is of great interest in generation tasks. It does so by computing the gradient flow of the entropy-regularized Sliced-Wasserstein (SW) distance in the *Wasserstein space*. The flow is simulated using links with well-known stochastic differential equations. For this project, I focus on the convergence of the SW gradient flow in low-dimensional spaces by computing the gradient explicitly. I study the impact of the number of sampled random directions to approximate the SW distance. I realize a non-exhaustive analysis of this heuristic by testing it on various distributions and on certain limit cases.

1 Introduction

Implicit generative modeling (IGM) [2] seeks to generate synthetic data samples that faithfully capture the features of a specified target data distribution. Variational auto-encoders (VAE) [3] and generative adversarial networks (GAN) have democratized these methods.
$\{y_1, \dots, y_P\}$ are independent and identically distributed (i.i.d.) observations sampled from the probability measure $\beta \in \mathcal{P}(\Omega)$, where $\mathcal{P}$ is the space of probability measures on the measurable space $(\Omega, \mathcal{A})$, $\Omega \subset \mathbf{R}^d$ is a domain, and $\mathcal{A}$ is the associated Borel σ-field. The IGM task associated with this setting consists in finding $T : \Omega_\alpha \to \Omega$ such that $T(x) = y$. Therefore, if x

comes from α a a well-known (and easy to sample from) probability measure on Ω_μ, $T(x)$ should fit the unknown target measure β on Ω and we would be able to generate new samples from β through the transport plan T.

Optimal transport [4] is precisely the study of transport plans between a starting measure α and a target measure β. We introduce the push forward operator on probability measure $T_\# : \mathcal{P}(\Omega_\alpha) \to \mathcal{P}(\Omega)$ which is defined as follows in our case: $\beta(A) = \alpha(\{x \in \Omega_\alpha; T(x) \in A\})$, for all Borel sets $A \subset \mathcal{A}$.

The paper of interest [1] proposes a new approach to finding this mapping function. Indeed, other methods such as the ones seen in *Probabilistic Graphical Models* class like VAE considers *parametric* families of transport map. In these models, introduced with Wasserstein distance by [5] the transport mapping can be approximated with a neural network. Then, the goal is to find the best parameters which enable to match the target distribution. There are learned from the observations and the fixed latent distribution which could be associated with the probability measure α here. However, these models lack convergence theoretical guarantees.

The studied paper aims at finding a model which iteratively generates ***non-parametric*** transport mapping with **theoretical guarantees**. Indeed, the authors provide finite-time bounds which is a non-asymptotic error guarantee and establish explicit links between the algorithm parameters and the overall error.
The method generates T_t at time $t \geq 0$ and we define the approximation of the target distribution after a time t: $\alpha_t = T_\#\alpha$. In order to force α_t to converge towards β the problem can be seen as finding the solution (β) of an optimization problem. In $\mathbf{R}$ an analogy would be then to find the minimum of a function, gradient-based methods such as gradient descent are standard methods to perform these kind of tasks. Similarly as finding a minimal point in $\mathbf{R}$ a gradient flow in the space of probability measure can be defined. In the paper, the following gradient flow is designed:

$$\partial_t \alpha_t = -\nabla_d \left(\mathrm{Cost}(\alpha_t, \beta) + \mathrm{Reg}(\alpha_t)\right) \tag{1}$$

where d denotes a metric for probability measures (2-Wasserstein distance for instance), *Cost* represents the divergence between α_t and β and *Reg* enables to regularize the generated distribution. Using the same analogy as before the *Cost* could be the distance between the two points and the regularization would regulate the value of the norm of the solution like for a Ridge Regression problem for instance.

According to [6], simulating this flow should lead $\alpha_t = T_\#\alpha$ to converge to the minimum of the functional optimization problem: $\min_\alpha(\mathrm{Cost}(\alpha, \beta) + \mathrm{Reg}(\alpha))$.

In the paper of interest, the sliced-Wasserstein distance [7] is used as the

Cost functional. It consists in a variation of the Wasserstein distance which leverages the fact that in the one-dimensional case the 2-Wasserstein distance has an analytical form which leads to a closed-form for the optimal transport mapping. This is essential since computing the 2-Wasserstein distance is computationnally costly. For instance, Sinkhorn algorithm solves a relaxed version of the Wasserstein distance (with an entropy regularization) in $\mathcal{O}(n^2)$ with n the number of components of the distributions in the discrete case. Furthermore, using the sliced-Wasserstein distance and the negative entropy as regularization function enable that the gradient flow can be simulated. They compute it by leveraging connections between a Partial Differential Equation (PDE), which should be solved by α_t, and a Stochastic Differential Equation (SDE) whose solutions can be computed.

To my understanding, the method this paper introduces stands out as a novel nonparametric approach in implicit generative modeling. The authors derive theoretical guarantees for the convergence of the α_t probability measure. Beyond its appealing theoretical aspects, the algorithm holds noteworthy practical value by demanding minimal computational resources.

For this project, I first visualize gradient flows in $2D$ and $3D$, then I realize an in-depth analysis of the convergence of the flow depending on the number of randomly sampled directions to approximate the integral on the unit sphere (necessary to compute the SW distance). Finally, I try to highlight limits of the method for particular distributions and I illustrate the benefits of the entropy-regularization (noise).

All the code and necessary resources can be found here [1].

2 Technical background

2.1 Wasserstein distance

Let's define a transportation plan from α to β , a coupling in the space of probability measures $\mathcal{P}(\Omega \times \Omega)$. It characterizes the amount of mass $\pi(x, y)$ to be moved from the distribution α at point x to point y, resulting in the creation of the distribution β in aggregate. The set of admissible transportation plans, denoted as $\mathbf{\Pi}(\alpha, \beta)$, is defined as follows:

$$\mathbf{\Pi}(\alpha, \beta) = \{\pi \in \mathcal{P}(\Omega \times \Omega) : \pi(\cdot, \Omega), \pi(M, \Omega) = \alpha, \beta\}.$$

This set captures transportation plans that satisfy mass conservation laws (incompressibility), ensuring that the transformation adheres to the specified source and target distributions.

[1] https://github.com/HipPilchen/Sliced-Wasserstein-Flows

After Gaspard Monge intuition, L. Kantorovich [8] was the first to formulate the optimal transport problem as an optimization problem. The objective is to minimize the average cost of transportation from one distribution to another on all the possible mapping plans:

$$\inf_{\pi \in \Pi} \left\{ \iint c(x,y) d\pi(x,y), \pi_1 = \alpha, \pi_2 = \beta \right\} \tag{2}$$

with the two values π_1 and π_2 which represents the incompressibility constraints and c the cost function, which is in this case the distance between the two points. In the setting of the studied paper, Ω is a compact of $\mathbf{R}^d$. The Kantorovitch problem above can be reformulated as a calculation of the Wasserstein distance. Therefore, for the 2-Wasserstein distance, the optimization problem for finding the optimal transportation becomes:

$$W_2(\alpha, \beta) = \left(\inf_{\pi \in \Pi} \int_{\Omega \times \Omega} \|x - y\|^2 \, d\pi(x,y) \right)^{\frac{1}{2}} \tag{3}$$

Furhtermore, in $\mathbf{R}^d$, if one of the two probability measures has a density (is absolutely continuous w.r.t Lebesgue measure) Kantorovitch and Monge problems are equivalent. It is the case here and also $\alpha, \beta \in \mathcal{P}_2(\Omega)$ with $\mathcal{P}_2(\Omega) = \{\alpha \in \mathcal{P}(\Omega) : \int_\Omega \|x\|^2 \mu(dx) < +\infty\}$ (with finite second-order moments). Therefore, Brenier's theorem [9] ensures that there exists a unique optimal transport mapping T, then the optimal transportation plan becomes: $\pi^* = (\mathrm{Id} \times T)$. It also states that the transport mapping is linked to the gradient of a Kantorovich potential ψ (it appears in the dual formulation of the 2-Wasserstein distance). Nevertheless, when dealing with continuous and high-dimensional probability measures, the task of constructing an optimal transport plan remains a challenge.

2.2 Wasserstein gradient flows

The following statements are derived from [10]. In an Euclidean space, given a function $F : \mathbf{R}^n \to \mathbf{R}$, smooth enough, and a starting point $x_0 \in \mathbf{R}^n$, a gradient flow is a curve $x(t)$ starting at $t = 0$ from x_0. The trajectory moves by choosing at each instant of time the direction which makes the function F decrease the most. More generally, given a metric space $(\mathcal{X}, d)$ and a lower semi continuous function $F : \mathcal{X} \to \mathbf{R}$ this curve can be reproduced by the following equation (time is here discretized but we can come back to continuous time by piecewise constant interpolation $t = \tau k$): $x_{(k+1)\tau} \in \mathrm{argmin}_x F(x) + \frac{d(x, x_k \tau)^2}{2\tau}$. τ represents the timestep and in the limit $\tau \to 0$ one could expect to recover something close to a gradient flow in the metric space $(\mathcal{X}, d)$. E. DeGiorgi in [11] defined this as a Generalized Minimizing

Movements. In the same way, the metric space can be $(\mathcal{P}(\Omega), W_2)$ and then the gradient flow equation becomes:

$$\alpha_{k+1} = \text{argmin}_{\alpha \in \mathcal{P}(\Omega)} W_2(\alpha, \alpha_k)^2 + \tau F(\alpha) \tag{4}$$

, with τ the implicit time step. This iterated minimization scheme is called Jordan–Kinderlehrer–Otto (JKO) scheme by [12].

This gradient flow exhibits strong connections with certain partial differential equations (PDEs) [10]. Specifically, it is demonstrated that under specific conditions on F, the family $(\alpha_t)_t$ is a solution of the gradient flow if and only if it possesses a density ρ_t with respect to the Lebesgue measure for all $t \geq 0$ and satisfies the continuity equation: $\partial_t \rho_t + \text{div}(v\rho_t) = 0$, where v represents a vector field, and div denotes the divergence operator. Subsequently, for a given gradient flow in $(\mathcal{P}_2(\Omega), W_2)$, the paper focuses on understanding the evolution of the densities ρ_t, i.e., the partial differential equations (PDEs) they adhere to.

2.3 Sliced-Wasserstein distance

1-D Wasserstein distances have an analytical form using the cumulative distribution functions:

$$W_2^{1D}(\alpha, \beta) = \left| \int_0^1 |C_\alpha^{-1}(t) - C_\beta^{-1}(t)|^2 dt \right|^{\frac{1}{2}} \tag{5}$$

, with C_α^{-1} the quantile function of the probability measure α. In this setting, the optimal solution for mapping α to β is $T = C_\beta^{-1} \circ C_\alpha$ (the first part maps α to $\mathbf{1}_{[0;1]}$ and the second maps $\mathbf{1}_{[0;1]}$ to β).

Therefore, the idea behind sliced Wasserstein distance is to link Wasserstein distance in high dimension to the one-dimensional one, in order to use the analytical form described above. The sliced Wasserstein distance [7] (I used notations of *Computational optimal transport* [4] , from Gabriel Peyré and Marco Cuturi, rather than the ones of the paper) is the aggregation of the 1-D Wasserstein distances between the projections of (α, β) (defined on $\mathbf{R}^d$) onto all directions of the unit sphere:

$$SW_2(\alpha, \beta) = \left| \int_{\mathbf{S}^d} W_2^{1D}(P_{\theta,\#}\alpha, P_{\theta,\#}\beta)^2 d\theta \right|^{\frac{1}{2}} \tag{6}$$

, where $\mathbf{S}^d$ is the d-dimensional unit sphere, P_θ the projection onto θ and $d\theta$ represents the uniform probability measure on $\mathbf{S}^d$. It is, indeed, a distance [13]. Therefore, minimizing the sliced Wasserstein distance comes down to find two equal probability measure $\alpha = \beta$. Furthermore, for measure supported on a ball both Wasserstein distance and sliced Wasserstein distance

are equivalent. If the projection can be computed, then both the SW_2 and the optimal transport map can be computed. Indeed, one can estimate (6) through a straightforward Monte Carlo approach, which involves generating uniform random samples from $\mathbf{S}^d$ and substituting the integral with an average over a finite sample.

3 The method: Sliced-Wasserstein Flows

3.1 Description

In the studied paper, the functional introduced in (2.1) is defined as follows:

$$F(\alpha) = \frac{1}{2} SW_2(\alpha, \beta)^2 + \lambda H(\alpha) \tag{7}$$

, where λ is the regularization parameter and H the negative entropy if α absolutely continuous with respect to the Lebesgue measure and ∞ otherwise. This regularization adds noise to the generated distributions. It enables to diversify the generated samples from the reconstructed probability measure α_t, avoiding to sample from the available observations of β.

The authors of the paper proved (Theorem 2 in [1]) that using this functional, in the case where β is a probability measure on the unit ball with a strictly positive smooth density, then there exists a flow α_t and the density of this measure, ρ_t for all $t > 0$, satisfies the following equations:

$$\frac{\partial \rho_t}{\partial t} = -\text{div}(v_t \rho_t) + \lambda \Delta \rho_t \tag{8}$$

$$v_t(x) = -\int_{\mathbf{S}^d} \psi'_{t,\theta}(P_\theta(x))\theta d\theta \tag{9}$$

, with ψ the Kantorovich potential (which I mentioned earlier in connection with Brenier's theorem) between $P_{\theta,\#}\alpha_t$ and $P_{\theta,\#}\beta$.

Brief overview of the connection with SDE and its implication

Thanks to the definition of the functional, the authors of the article exhibit connections between the continuity equations of the flow (9) and well-known solvable Stochastic Differential Equation (SDE). I will briefly summarize these links here to help understand the theoretical guarantees that follow from them. The PDE (9) is a Fokker-Planck-type equation [14] which can be expressed as the following SDE: $dX_t = v(X_t, \alpha_t)dt + \sqrt{2\lambda}dW_t$, with $(W_t)_t$ a Brownian motion. Then, drift, v, dependancy in X_t leads the authors to use a collection of SDEs which approximate the SDE above. They define the following *particle system*: $dX_t^i = v(X_t^i, \alpha_t^N)\, dt + 2\lambda\, dW_t^i, \quad i = 1, \ldots, N$, where i is the particle index, N the number of particles and α_t^N the empirical distribution of the N particules $X^i_{t\, i\in[1;N]}$ at time t. Each of these SDEs

can be solved using an Euler-Maruyama discretization scheme (time t is discretized). The algorithm iteratively applies the following update equation, for each particle i:

$$X_0^i \overset{i.i.d}{\sim} \alpha_0, \quad X_{k+1}^i = X_k^i + h\hat{v}_k(X_k^i) + \sqrt{2h\lambda}Z_{k+1}^i \tag{10}$$

where k denotes the iteration number, Z_k^i is a normal random vector in $\mathbf{R}^d$, h denotes the step-size, and $\hat{v}_k$ is a tractable estimator of the original drift at time kh. Then, by approximating the integral in (9) for the drift by a Monte Carlo method this iterative process leads to an empirical distribution α_{Kh}^N which converges toward β. The cumulative distribution functions and the quantiles functions are empirically approximated.

3.2 Theoretical guarantees

The first theoretical guarantee comes from **Theorem 2**[1]. It states that **the gradient flow** defined in the paper ($\min_\alpha F(\alpha)$ with F from (2.1)) **exists** and that there exists a *minimizing movement scheme* $(\alpha_t)_t$. **Continuity equations** should be verified by its associated family of densities $(\rho_t)_t$ for all t. This means that the family of probability measures, associated with the family of densities $(\rho_t)_t$ which solve the continuity equation (9), decreases along F (2.1). The algorithm described in the paper aims at solving this continuity equation.

Then, the second main theoretical result from **Theorem 3** and **Corollary 1** [1] ensures that the approximated solution computed with the Sliced-Wasserstein Flow (SWF) algorithm gets close to the true solution of the continuity equation at the same time (with the total variation as distance). It is a **non-asymptotic error guarantee** for a step-size h small enough (10). The bound on the distance between the two probability measures depends on $1/\sqrt{\lambda}$ and is proportional to the variance of the estimated drift multiplied by the time we are looking at. This means that at fixed time t increasing the entropy (by increasing λ) decreases the error.
However, the solution of (9) might then be far from the target distribution β since too much noise is added. Furthermore, in order to keep a low error approximation at large time the variance of the estimated drift should remain low.

To summarize, in [1], the authors proved the existence of $(\alpha_t)_t$ which decreases along (7). Then, they proved a non-asymptotic error guarantee for their method to find this solution $(\alpha_t)_t$. Consequently, the SWF (Sliced-Wasserstein Flow) algorithm calculates a distribution close to the target distribution while avoiding overfitting thanks to entropy regularization.

4 Experiments

To perform experiments with this sliced Wasserstein flow, I focus on the non-regularized version of the functional (the sliced Wasserstein distance only). Furthermore, I restrict myself to distributions in small spaces to reduce running time and facilitate the finding of interesting insight about convergence. I was able to establish this direction in my experiments thanks to the instructions and discussions with Prof. Gabriel Peyré, who helped me a great deal.
Gradient flow presented in (4) is solved by implicit stepping. Such methods have been summarized in [15]. They sometimes require entropy-regularization to relax the problem (like in [16]). After extensive research and discussions with my teacher, I decided to simulate this gradient flow with an explicit approach.

The Sliced-Wasserstein distance is approximated as follows:

$$SW_2(\alpha,\beta)^2 = \frac{1}{N}\sum_{k=1}^{N} W_2^{1D}(P_{\theta_k,\#}\alpha, P_{\theta_k,\#}\beta)^2 \tag{11}$$

, where N is the number of directions uniformly sampled from $\mathbf{S}^d$. Then, the explicit gradient on a sampled point of α (α_i) becomes

$$\frac{1}{N}\sum_{k=1}^{N}(< \theta_k, \alpha_i > - < \theta_k, \beta_{\sigma^k(i)} >)\theta_k$$

Here, σ^k is the permutation which maps each data point projected on direction k, namely $< \theta_k, \alpha_i >$, to the projected point of the target distribution which is at the same index in the sorted list of target samples projected along the same direction. This optimal mapping is also known as the *increasing arrangement*, which maps each quantile of α to the same quantile of β, e.g. minimum to minimum, median to median, maximum to maximum. In this special case where distributions are discretized, this mapping consists in sorting and matching points in the same order. This matching corresponds to the optimal transport plan in $1D$. To implement this, I first project all the samples along the chosen number of random directions on the unit sphere. These directions were sampled using samples from a normal distribution which have been normalized. Then, I sort all the projected samples along each direction (from both distributions) and finally I match samples at the same index for each direction. I reproduce the minimizing scheme by performing a gradient descent on a cloud of points (sampled from a precise distribution) using the gradient described above. To avoid exploding or vanishing gradient, I normalize the calculated gradient at each iteration.

4.1 Flows visualization and convergence

In this part, I show different visualization of empirical distributions convergence using explicit gradient flow with the Sliced-Wasserstein functional. I set the number of directions to 1000 which in practice is high enough to enable convergence. The impact of the number of directions on the convergence is discussed in the next section.

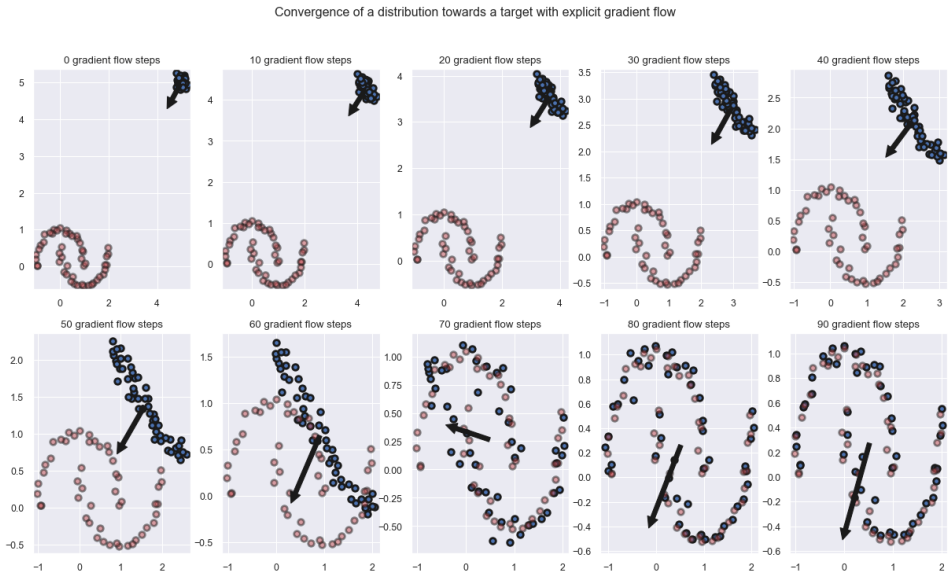

Figure 1: 2D visualization of the gradient flow starting with samples from a gaussian distribution (blue) toward a moon-shaped distribution in 2 dimensions (red). The arrow illustrates the average direction of the gradient. Both clouds have 50 points and the step-size in the gradient descent is 2.

Visually, the distribution approaches the moon-shaped distribution after a few steps of gradient flow. However, to confirm convergence I used the 2-Wasserstein distance as a metric to quantify convergence. To compute this distance I used *CVXPY*[2] library to solve optimal transportation of discrete distributions using the 2-norm in the cost matrix. As shown in Figure 2, the distance between the two empirical distributions decreases throughout the gradient steps until it reaches a value near 0.

[2] https://www.cvxpy.org

Figure 2: Plot of the 2-Wasserstein distance between the cloud from the moon-shaped distribution and the other cloud of points on which gradient steps are performed. The parameters are the same as in Figure 1.

Similarly, in higher dimensional spaces, the gradient flow converges toward the target distribution. The visualization of two $3D$ empirical distributions is displayed in Figure 3. However, despite efforts to make calculations faster, each gradient calculation is significantly lengthened when one of the parameter is increased. Indeed, if we denote d the dimension, N the number of random directions sampled and n the number of samples of the empirical distributions, one iteration of gradient flow has a complexity of $\mathbf{O}(N * (n \log n + nd))$. Therefore, experiments were mostly done in rather low-dimensional spaces and with not so many samples.

In $3D$ the two empirical distributions might not be visually identical. However, the 2-Wasserstein distance enables to evaluate the disparities of these two cloud of points and ensures that the gradient flow has indeed converged towards the targeted distribution (or at least close, $W2 \approx 0.001$).

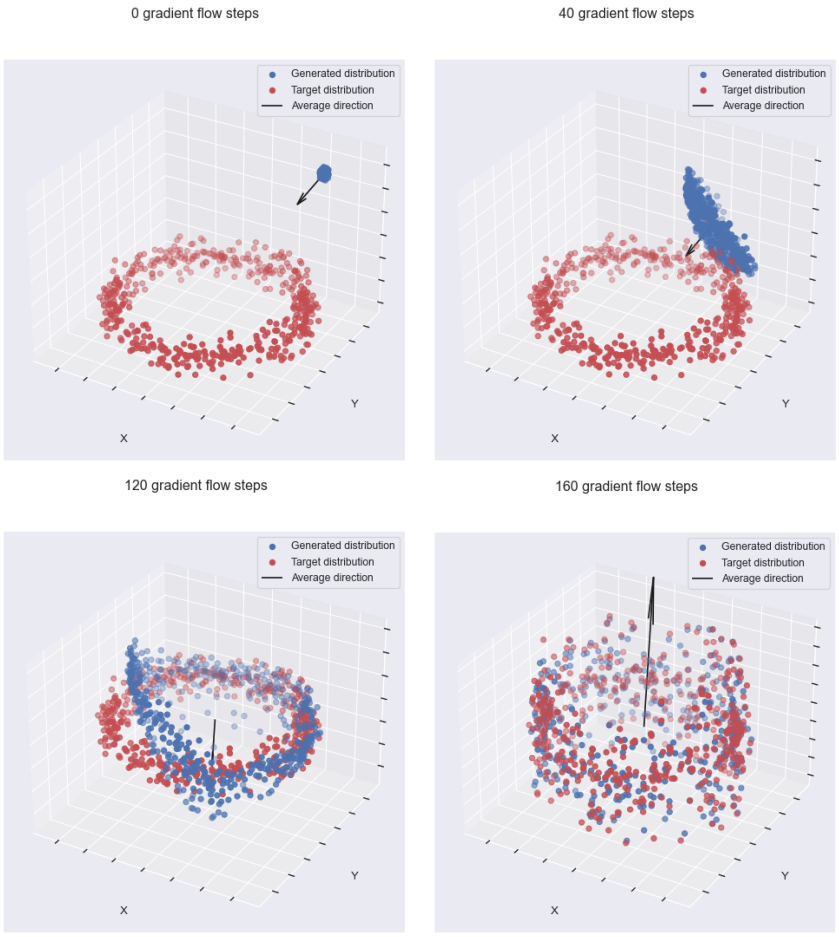

Figure 3: 3D visualization of the gradient flow starting with samples from a gaussian distribution (blue) toward a torus-shaped distribution in 3 dimensions (red). The arrow illustrates the average direction of the gradient. Both clouds have 500 points, the step-size in the gradient descent is 1 and 500 directions were used. The scale of the axis changes between different plots

4.2 Impact of the number of directions

To compute the Sliced-Wasserstein gradient we need to sample random directions on the unit sphere of the studied space and then project the data points (samples from studied distributions) along these directions. Even though convergence of the flow should be linked with the dimension of the space, due to computation time, I study the impact of the number of directions on the convergence of the flow in a $3D$ space with a gaussian target distribution. Settings of the experiment are illustrated in Figure 4.

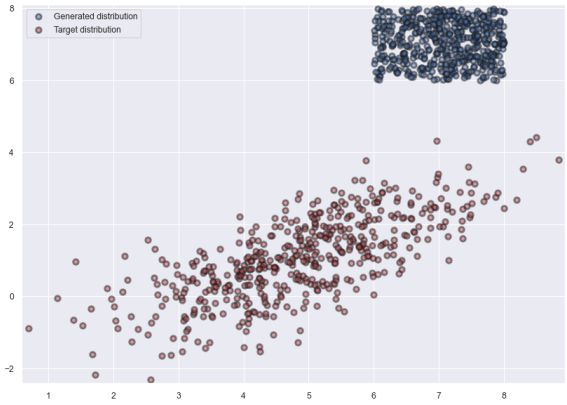

Figure 4: $2D$ visualization of the starting and targeted cloud of 3-dimensional points. The starting distribution from which we sample is a shifted and dilated uniform distribution on $[0, 1]^3$. The target distribution is gaussian distribution. 500 points are sampled for each distribution and the step-size of the gradient flow is set to 0.1.

The results of this experiment are displayed in Figure 5. The number of directions does not impact the stopping of the gradient flow. It affects the local minima in which the flow gets stuck. According to [13], there exists an equivalence equality between the Sliced-Wasserstein distance and the 2-Wasserstein distance. Since the global minimum of SW distance is 0, if the flow converges towards the target we should obtain a final 2-Wasserstein distance of 0. The step-size can prevent from reaching exactly this value, however we can see that by increasing the number of directions we can get closer to the target distribution.

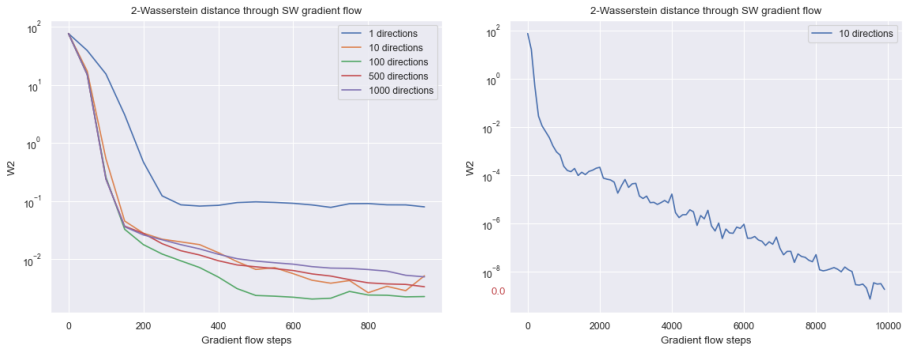

Figure 5: Plot of the 2-Wasserstein distance during gradient flow in semi-log scale: (left) for various number of directions with fixed step-size and (right) for 10 directions with a decreasing step-size. The other parameters are the same as in Figure 4

Increasing the number of directions enables to obtain a faster convergence of the gradient flow. However, it seems that there is a threshold number of directions above which the gradient flow behavior remains identical. Here, this threshold lies between 1 and 10 directions, since for numbers of directions greater than 10 we observe the same convergence curve. For number of directions below this threshold, it seems that the gradient flow gets stuck in local minima while above this the flow converges towards the target distribution. Indeed, the plateau reached for 10 directions is due to the step-size and by slowly decreasing this step-size the gradient flow converges exactly (distance below $10^{-}10$ in right part of Figure 5). Two phases can be underlined during convergence: first a rapid decrease which consists in bringing the cloud of points around the same mean that the target and then a slower decrease which consists in reshaping the cloud of points to match the target distribution.

Visually both point clouds quickly become difficult to distinguish ($\approx$ 200 iterations), but the gap between 1 direction and the other experiments remains visible (see Figure 8 in Appendix A).

Limitations

However in certain case this convergence seems empirically difficult. For multimodal distributions, the gradient flow seems to stop at the barycenter of the target cloud of points (Figure 6). The gradient flow gets stuck in rather high energy local minima and an increasing number of directions does not solve this issue. Indeed, the behavior of the gradient flow seems identical for 10000 sampled directions and with various policies of decreasing step-size.

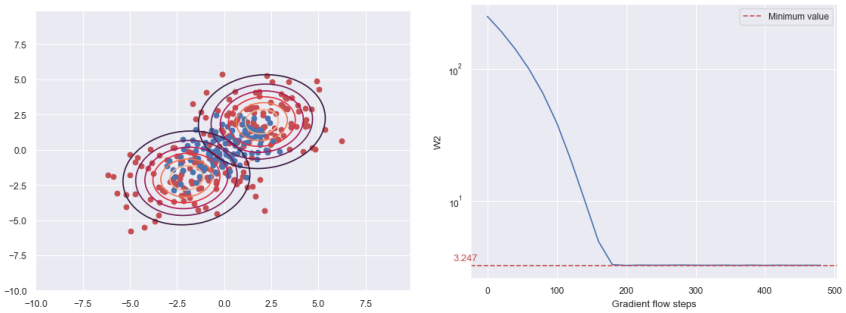

Figure 6: $2D$ visualization of the converged (blue) and targeted (red) cloud of 2-dimensional points and plot of the 2-Wasserstein distance. The starting distribution from which we sample is a shifted and dilated uniform distribution on $[0, 1]^2$. The target distribution is a mixture of gaussian (a bimodal distribution). The contour are drawn from the parameters of the two gaussian distributions. 100 points are sampled for each distribution, 10000 directions were sampled and the step-size of the gradient flow was set to 0.1.

The convergence speed of the flow decreases through iterations, nevertheless, it still continues to slightly decrease the 2-Wasserstein distance after reaching the plateau (as seen in Figure 5 in semi-log scale). So I am conjecturing that for a high enough number of directions, flow should be able to reach the target but in a rather large number of iterations.

4.3 Simulation of noisy gradient flow

As explained in the paper [1], the noise introduced by the negative entropy penalty enables to fit the targeted distribution while including noise to avoid overfitting. It leads to probable samples from the targeted distribution (according to the observed samples) which are not in the training set. In generative modelling enabling such diversity is crucial. Therefore, I try in this section to add to the explicit gradient flow a gaussian noise and evaluate the probability and diversity of these new samples. I carry out this experiment with fewer samples for visual purpose but also because diversity (while guaranteeing accuracy) of the generation is crucial especially in domain with few samples like with medical data.

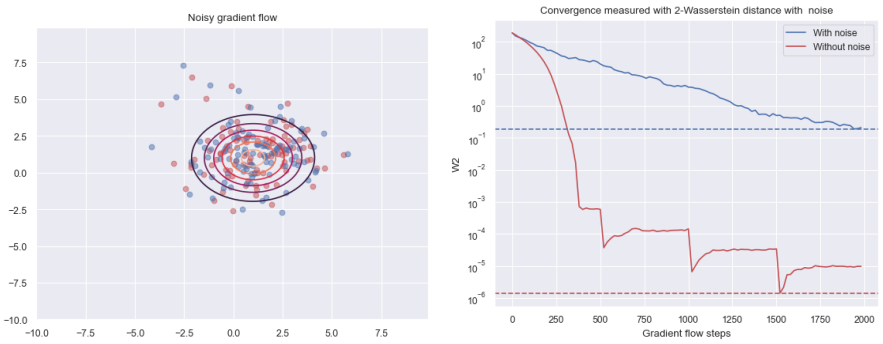

Figure 7: 2D visualization (left) and plot of the 2-Wasserstein distance (right) for a gradient flow targeting 100 samples of a gaussian distribution. The step-size decreases during learning and a gaussian noise is added to the flow. The initial distribution is a shifted and dilated uniform distribution on $[0, 1]^2$ and 1000 directions are sampled.

The results in Figure 7 are pretty satisfying. Indeed, we can see that by adding a certain amount of noise to the flow a cloud of points, which concentrates in the area of high-probability of the targeted distribution, can be obtained. The 2-Wasserstein distance between the samples illustrates that the two cloud of points are different while the $2D$ visualization shows that the samples could follow the same distribution as the target. This could highlight the diversity of the SWF method in the generation of samples. However, these results should be treated with caution since the target distribution (gaussian) is symmetric around the barycenter of the cloud of points which could help the gradient flow.

5 Conclusion and perspective

Finally, I realize an analysis of the Sliced-Wasserstein flow by simulating this flow with an explicit gradient rather than using the paper's algorithm SWF. However, it highlights general behavior of the method which should be observed for the SWF algorithm also. It has been empirically shown that the gradient flow of the SW distance converges towards target distributions. The convergence has been observed for several distributions (Gaussian, moon-shaped, torus-shaped...) in different spaces. However, the gradient flow seems to fail to learn multi-modal distributions. This limitation could hinder the algorithm results on real data. Furthermore, the calculation of the explicit full gradient flow is computationally costly. There is no study of computational complexity in the paper but even though the algorithm to simulate the flow is different from the one I implemented I suspect that the SWF algorithm does not scale well in dimension and number of training

samples. The fact that testing on real data requires an auto-encoder may be an hint in this direction, indeed in higher dimension one need to sample more directions on the unit sphere.
However, on the theoretical side, the authors provide theoretical guarantees on the existence of the minimization scheme and on the non-asymptotic gap error. Moreover, I tested an implementation to add noise to simulate the entropy-regularization. It leads to new samples which are in the high-probability area of the targeted distribution while not being close to the training cloud of points. It is an interesting feature for generative models. Furthermore, images generated with SWF (in the paper) are comparable to the one generated by other generative models (like VAEs) which illustrates the accuracy of the method. To conclude, I have tried to illustrate that the SWF method enables in most of the case to accurately converge towards a distribution from which we only have a few samples. This generative model benefits from explicit theoretical properties which is uncommon in the generative modeling literature.

6 From Computational Optimal Transport class:

During this project I used several notions and results from the *Computational Optimal Transport* class. First, the various metrics and operators introduced in the course like the push forward operator are used in the method's framework. Secondly, the $1D$ general case of the general Monge problem helps me to understand why the authors choose to use Sliced-Wasserstein as functional. Furthermore, Brenier's theorem enables to derive the continuity equations which are at the center of the resolution of the Sliced-Wasserstein flow. Finally, the brief introduction to density fitting problem at the end of the course gave me a basis to compare the paper [1] with other generative models using optimal transport.

References

[1] Antoine Liutkus, Umut Şimşekli, Szymon Majewski, Alain Durmus, and Fabian-Robert Stöter. Sliced-wasserstein flows: Nonparametric generative modeling via optimal transport and diffusions, 2019.

[2] Shakir Mohamed and Balaji Lakshminarayanan. Learning in implicit generative models, 2017.

[3] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022.

[4] Gabriel Peyré and Marco Cuturi. Computational optimal transport, 2020.

[5] Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schoelkopf. Wasserstein auto-encoders, 2019.

[6] Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. Gradient flows: In metric spaces and in the space of probability measures. 2005.

[7] Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister. Sliced and radon wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51(1):22–45, 2015. hal-00881872.

[8] Lev Kantorovich. On the translocation of masses. *Journal of Mathematical Sciences*, 133:1381–1382, 2006.

[9] Yann Brenier. Polar factorization and monotone rearrangement of vector-valued functions. *Communications on Pure and Applied Mathematics*, 44:375–417, 1991.

[10] Filippo Santambrogio. {Euclidean, metric, and Wasserstein} gradient flows: an overview. *Bulletin of Mathematical Sciences*, 7(1):87–154, 2017.

[11] E. DeGiorgi. *In E.Lions, J. L. and Baiocchi, C. and Magenes, E. (eds.) Boundary Value Problems for Partial Differential Equations and Applications: Dedicated to E. Magenes.* Recherches en mathématiques appliquées. Masson, 1993.

[12] Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the fokker–planck equation. *SIAM Journal on Mathematical Analysis*, 29(1):1–17, 1998.

[13] Nicolas Bonnotte. *Unidimensional and Evolution Methods for Optimal Transportation.* Theses, Université Paris Sud - Paris XI ; Scuola normale superiore (Pise, Italie), December 2013.

[14] V.I. Bogachev, N.V. Krylov, M. Röckner, and S.V. Shaposhnikov. *Fokker–Planck–Kolmogorov Equations.* Mathematical Surveys and Monographs. American Mathematical Society, 2022.

[15] Clément Bonet, Nicolas Courty, François Septier, and Lucas Drumetz. Efficient gradient flows in sliced-wasserstein space, 2022.

[16] Gabriel Peyré. Entropic wasserstein gradient flows, 2015.

A Additional visualizations

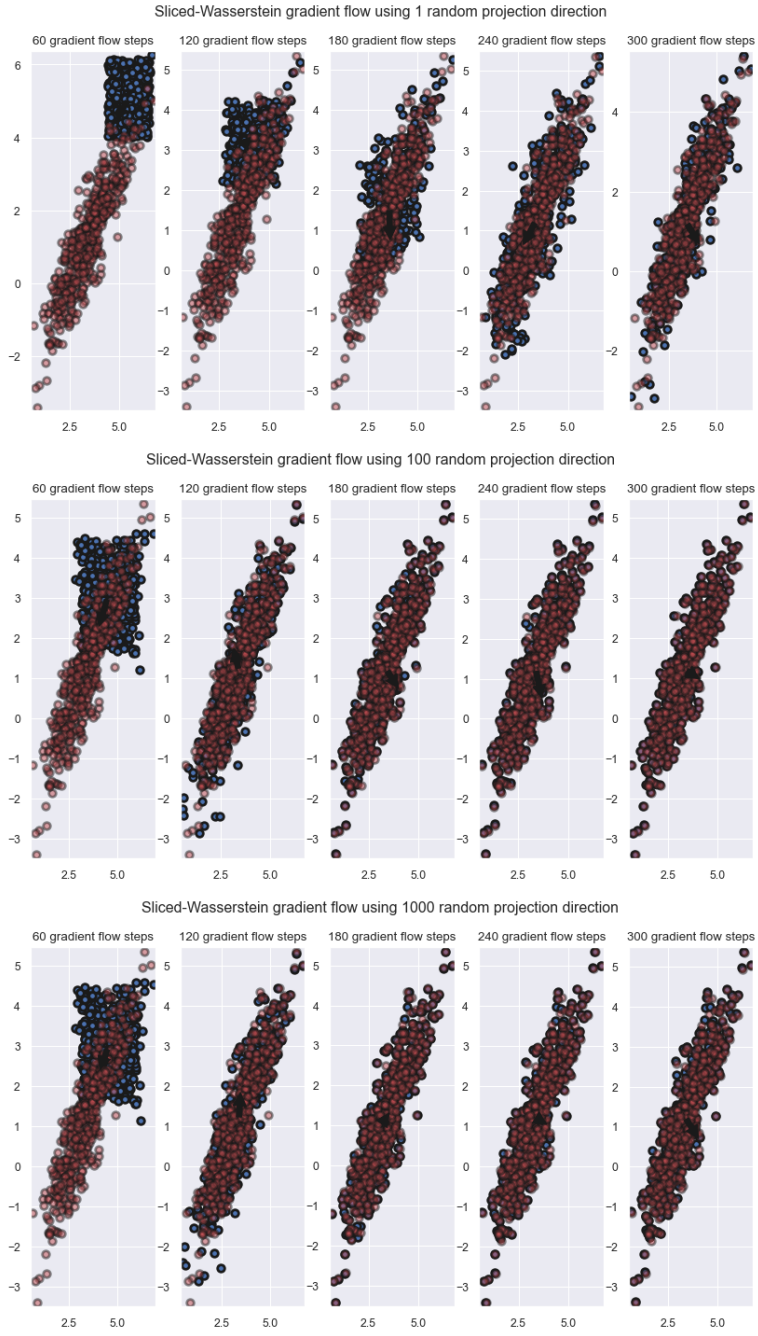

Figure 8: $2D$ visualization of the gradient flow for several number of directions. The starting distribution from which we sample is a shifted and dilated uniform distribution on $[0, 1]^3$. The target distribution is gaussian distribution. 500 points are sampled for each distribution and the step-size of the gradient flow is set to 1.