Project report: Learning with minibatch Wasserstein: asymptotic and gradient properties

Quentin Bourbon

January 16, 2024

Abstract

This report summarises the main contributions of the paper [3] on the use of mini-batch when dealing with optimal transport distances. Indeed, distances derived from optimal transport can be useful in many applications in machine learning, but the time complexity to compute them prevents them from being really used in practice. Then a natural way to reduce the time complexity is to divide the initial problem into several smaller problems and average the total. Therefore, [3] proposes to analyse this method in depth. In particular, he shows that this method is equivalent to a regularisation of the initial problem, with more attractive properties than the initial problems, such as unbiased estimators, gradients, and a concentration bound around the expectation. Hence, his contribution is more theoretical around the practices used in machine learning problem.

1 Introduction

Measuring the distance between two sets of points that follow probability distributions is a crucial point in machine learning problems. Indeed, in supervised learning, the loss function is mainly defined around the distance between the source data and the target. For this reason, optimal transport has emerged in machine learning because it provides a very natural and canonical distance to optimise.

Then, the Wasserstein distance between two measure distributions $\alpha \, in \mathbb{M}(\mathcal{X})$ and $\beta \in \mathbb{M}(\mathcal{X})$ is defined with respect to a cost function c between α and β, such as

$$W_c(\alpha, \beta) = \min_{\pi \in U(\alpha,\beta)} \int_{\mathcal{X}\times\mathcal{X}} c(x, y) d\pi(x, y) \tag{1}$$

where $U(\alpha, \beta)$ is the set of joint probability distributions with marginals α and β such that $U(\alpha, \beta) = \{\pi \in \mathbb{M}(\mathcal{X}, \mathcal{X}) | P_{\mathcal{X}\#\pi} = \alpha, P_{\mathcal{Y}\#\pi} = \beta\}$. It

ensures the conservation of mass of the distributions in transport (no loss or creation of density in α or β). When we deal with discrete measures, as we do naturally in machine learning, (1) becomes

$$W_c(\alpha, \beta) = \min_{\pi \in U(\alpha,\beta)} \sum_{i,j} c(x_i, y_j)\Pi_{x_i,y_j} \tag{2}$$

Π is known as the transport plan. It allows us to go from the α distribution to the β distribution. For simplicity, we will consider the particular case where the number of samples n in α is equal to the number of samples in β, and where all weights are uniform. Thus, the admissible set $U(\alpha, \beta)$ in this case becomes: $U(\alpha, \beta) = \{\Pi \in \mathbb{R}_+^{n \times n} | \Pi \mathbf{1}_n = \mathbf{1}/n, \Pi^T \mathbf{1}_n = \mathbf{1}/n\}$. The set of matrices $U(\alpha, \beta)$ is bounded, defined by $2n$ equality constraints, and is thus a polytope. Finding the optimal transport plan by solving (2) is thus a convex linear program that could be solved in $\mathcal{O}(n^3 log(n))$ using the simplex algorithm.

Despite the fact that this algorithm is polynomial, it is too long for many cases, and one may want to speed up the solution of (2) even if it means not getting the exact result. Therefore, in many cases an approximation of the solution is enough to see if the algorithm is getting better or not. Therefore, [2] proposes a regularised entropic optimal transport that allows a solution in $\mathcal{O}(n^2)$. He defines entropic loss as

$$W_c^\epsilon(\alpha, \beta) = \min_{\pi \in U(\alpha,\beta)} \int_{\mathcal{X} \times \mathcal{X}} c(x, y) d\pi(x, y) + \epsilon \mathrm{KL}(\Pi | \alpha \otimes \beta)$$

where KL is the Kullback-Leibler divergence.

One problem with entropic regularisation is that W_c^ϵ is not a metric, since $W_c^\epsilon(\alpha, \alpha) \neq 0$. So [5] introduced its unbiased version, called the Sinkhorn divergence, defined as

$$S_c^\epsilon(\alpha, \beta) = W_c^\epsilon(\alpha, \beta) - \frac{W_c^\epsilon(\alpha, \alpha) + W_c^\epsilon(\beta, \beta)}{2}$$

Varying the regularisation parameter ϵ allows to interpolate between OT (if $\epsilon \to 0$) and maximum mean discrepancy (if $\epsilon \to +\infty$) as shown in [4]. However, the classical Sinkhorn algorithm, which allows to compute the solution in $\mathcal{O}(n^2)$, is unstable when ϵ is too small. Then one can compute the solution with a stabilised Sinkhorn algorithm (when dealing with the dual problem), but this makes the algorithm longer. Therefore, in this report we will stick to the classical Sinkhorn algorithm, taking care to choose an appropriate ϵ.

However, machine learning deals with a lot of data, which makes computation impossible in practice. This is a well-known problem in machine learning, and the solution is to compute the value of interest not on all the data, but on a batch of data. Thus, one can adapt the same procedure to

compute the Wasserstein distance on batches of data. [3] deals with the justifications and some properties of this use. They first correctly defined the notion of minibatch Wasserstein distance, with the subsampling version. They correctly defined the transport plan associated with the minibatch distance. They studied basic properties of minibatch distance and transport plan, the convergence of the subsampling version, and concluded with a gradient property that allows the use of SGD using minibatch Wasserstein. My own contribution is to better interpret the behaviour of the minibatch transport plan when the batch size and the number of draws vary.

2 Minibatch Wasserstein

In this section we first define the minibatch Wasserstein as defined in [3]. Then we present the results obtained in [3] on asymptotic properties and optimisation behaviour.

2.1 Notations and Definitions

Let $\mathbf{X} = (X_1, ..., X_n)$ (resp. $\mathbf{Y} = (Y_1, ..., Y_n)$) be samples of n iid random variables drawn from a distribution α (resp. β) on the source (resp. target) domain. We denote by α_n and β_n the empirical distributions of support $\{X_1, ..., X_n\}$ and $\{Y_1, ..., Y_n\}$ respectively. We assume that the weights of X_i and Y_i are uniformly equal to $\frac{1}{n}$. In additive, we assume that α and β have compact support, so that the ground cost is bounded by a constant M.

To deal with mini-batches, we denote by $\alpha^{\otimes m}$ a sample of m random variables following α. Note that a batch A of cardinality m is associated with the distribution $\hat{A} := \frac{1}{m}\sum_{a\in A}\delta_a$. We also recall that there are exactly $\binom{n}{m}$ lots of size m when we consider n samples. We call $\mathcal{P}_m(\alpha_n)$ the set of all lots of size m if α_n: $\text{Card}(\mathcal{P}_m(\alpha_n)) = \binom{n}{m}$.

Every quantity we will deal with is related to an OT distance/loss/divergence. All results (except when precise) are valid by choosing your favourite measure: Wasserstein Distance, Entropic Loss, Sinkhorn Divergence. We will denote one of these quantities by h-loss or OT-loss.

Definition 1 (Minibatch Wasserstein)**.** *Given an OT loss h and an integer $m \leq n$, we define the following quantities:*

The continuous loss:

$$U_{h,m}(\alpha, \beta) = \mathbb{E}_{(X,Y)\sim\alpha^{\otimes m}\otimes\beta^{\otimes m}}[h(X, Y)] \tag{3}$$

The semi-discrete loss:

$$U_{h,m}(\alpha, \beta) = \binom{n}{m}^{-1} \sum_{A\in\mathcal{P}_m(\alpha_n)} \mathbb{E}[h(X, Y)] \tag{4}$$

The discrete-discrete loss:

$$U_h(\alpha_n, \beta_n) = \binom{n}{m}^{-2} \sum_{A \in \mathcal{P}_m(\alpha_n)} \sum_{B \in \mathcal{P}_m(\beta_n)} h(A, B) \tag{5}$$

These quantities are, of course, defined as the average of the h-loss over a batch of size m. Depending on the value of m we use, we can hope that these quantities are a very good approximation of the true h-loss. Note that when $m = n$ we find back the true h-loss quantity. On the other hand, when $m = 1$, it becomes just the sum of the costs between all possible pairs of points. However, when n becomes too high, the minibatch Wasserstein is the sum of the h-loss between two sets of m points a huge number of times. In practice, when $n \geq 25$, the computation of the exact minibatch Wasserstein is too expensive. For this reason one can decide to take only k sets of m points (with $k << \binom{n}{m}$) and average the h loss over them. This gives the following definition.

Definition 2 (Minibatch subsampling)**.** *Given an OT loss h and an integer $m \leq n$, we define the following quantity:*

$$\tilde{U}^k_{h,m}(\alpha_n, \beta_n) = k^{-1} \sum_{(A,B) \in D_k} h(A, B) \tag{6}$$

where D_k is a set of cardinality k whose elements are drawn at random from the uniform distribution on $\Gamma = \mathcal{P}_m(\{X_1, ..., X_n\}) \times \mathcal{P}_m(\{Y_1, ..., Y_n\})$

This quantity exists to approximate the minibatch Wasserstein $U_{h,m}(\alpha_n, \beta_n)$ that exists to approximate the ground truth $h(\alpha_n, \beta_n)$. It's important to note that this set is not deterministic and relies on the set D_k, which is stochastic. However, [3] has shown a very interesting deviation bound between $\tilde{U}^k_{h,m}(\alpha_n, \beta_n)$ and $\tilde{U}_{h,m}(\alpha_n, \beta_n)$ which is not deterministic. Thus, for sufficiently high values of k, $\tilde{U}^k_{h,m}(\alpha_n, \beta_n)$ is a very good approximation of $\tilde{U}_{h,m}(\alpha_n, \beta_n)$.

Transport plans are derived from h-loss, so here we do the same by defining transport plans derived from minibatch OT loss.

Definition 3 (Minibatch transport plan)**.** *Consider α_n and β_n two discrete probability distributions. For each $A = \{a_1, \ldots, a_m\} \in \mathcal{P}_m(\alpha_n)$ and $B = \{b_1, \ldots, b_m\} \in \mathcal{P}_m(\beta_n)$, we denote by $\Pi_{A,B}$ the optimal plan between the random variables, considered as an $n \times n$ matrix where all entries are zero except those indexed in $A \times B$. We define the averaged mini-batch transport matrix:*

$$\Pi_m(\alpha_n, \beta_n) = \binom{n}{m}^{-2} \sum_{A \in \mathcal{P}_m(\alpha_n)} \sum_{B \in \mathcal{P}_m(\beta_n)} \Pi_{A,B} \tag{7}$$

Similar to the minibatch OT loss, the minibatch transport plans is simply the average of all transport plans computed with sets of m points. We will see later that this defines a transport plan well, provided that we scale the minibatch OT matrices so that the sum along a row and a column is equal to $\frac{1}{n}$ and note $\frac{1}{m}$ to preserve conservation of mass. In practice, we compute the OT matrices between the sets of m points with our favourite algorithm and scale this matrix by $\frac{m}{n}$.

Following the idea of subsampling, it is very natural to define the subsampled minibatch transport plan based on minibatch subsampling.

Definition 4 (Subsampled minibatch transport plan)**.** *With the same notations in 3, we define the subsampled minibatch transportation matrix for A and B:*

$$\Pi_m^k(\alpha_n, \beta_n) = k^{-1} \sum_{(A,B)\in D_k} \Pi_{A,B} \tag{8}$$

where D_k is drawn as in 3

In general, this does not define a transport plan, since firstly nothing guarantees that every point of the data set will be sampled. However, we will see later that $\Pi_m^k(\alpha_n, \beta_n)$ converges to $\Pi_m(\alpha_n, \beta_n)$, so for high values of k we have good hopes of getting something close to an OT matrix.

2.2 Illustrations on simple examples

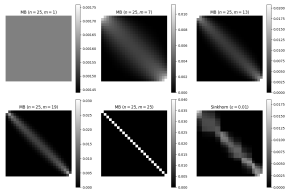
To illustrate the effect of the minibatch, we compute Π_m on two simple examples. We use the Wasserstein distance as h-loss to get an explicit formula in the 1D case.

Distributions in 1D The 1D case is very singular in the sense that we have a formula to map two discrete distributions using the Wasserstein distance. In fact, it is well known that for any cost of the form $C_{i,j} = h(x_i - y_j)$, where h is positive and convex, the mapping α to β comes only to sort the samples and to map samples with the same index ([6] for the proof). In terms of Π_n, this means that after sorting X and Y, Π_m is the identity matrix scaled by $\frac{1}{n}$ to preserve mass conservation. Then, using some combinatorial tricks and calculus, [3] has shown an explicit formula for Π_m when X and Y are sorted.

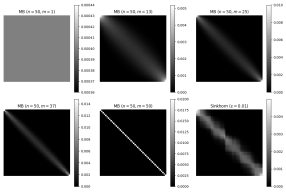
$$(\Pi_m)_{j,k} = \frac{1}{m}\binom{n}{m}^{-2} \sum_{i=i_{min}}^{i_{max}} \binom{j-1}{i-1}\binom{k-1}{i-1}\binom{n-j}{m-i}\binom{n-k}{m-i}$$

where $i_{min} = \max(0, m-n+j, m-n+k)$ and $i_max = \max(j,k)$ are the sorting constraints.

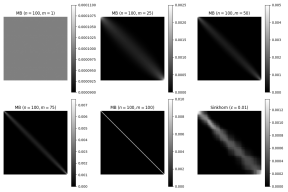
Using this formula, we obtain OT matrices between two uniform distributions of different sizes, shown in 1.



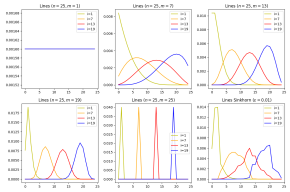
(a) OT matrix with $n = 25$



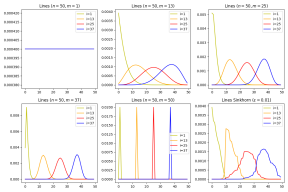
(b) OT matrix with $n = 50$



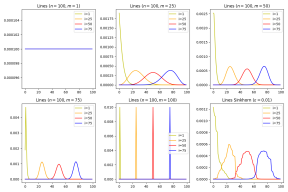
(c) OT matrix with $n = 100$



(d) OT matrix with $n = 25$



(e) OT matrix with $n = 50$



(f) OT matrix with $n = 100$

Figure 1: OT matrices between distributions in 1D depending on the number of samples n

We change the size n and proportionally the batch size m. We observe that the OT matrices become denser as m increases, like a kind of regularisation. We find the identity matrix when $m = n$, because it comes to the calculation of the classical Wasserstein OT matrix.

Plotting some rows of the OT matrices shows the spread of the mass of a point corresponding to the row. We can clearly see that the mass scatter is proportional. This shows a very elegant property of the minibatch Wasserstein to maintain proportionality as n increases.

Distributions in 2D The 2D case is more difficult because there is no explicit formula for the general case. Since the calculation of the minibatch transport plan Π_m requires the calculation of $\binom{n}{m}$ transport plans, n cannot be too large. So we have to limit ourselves to the case $n = 10$. To get the optimal plan a diagonal matrix, we first compute the true Wasserstein transport plan between, then we reorder the sets so that the first element of the first set maps into the first element of the second set... With this manipulation we obtain the identity matrix as a transport plan. We show our results in 2

We observe a lot of dispersal comparing to the 1d case and a bigger m seems to be more required than in 1d case.

Comparison with the subsampled version Here, we compare the minibatch Wasserstein with its subsampled version. Note that the subsampled transport plan is the result of a stochastic sampling so the results are not the always the same. The goal of this section is to compare the general behavior of Π_m^k on simple examples and not to study the asymptotic

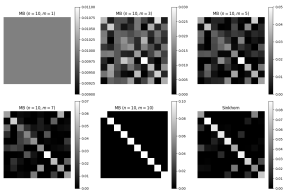
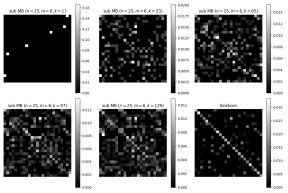


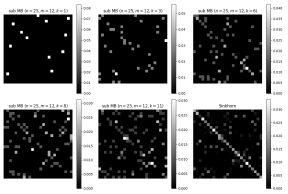
Figure 2: OT matrices between distributions in 2D depending on the batch size m

behavior of Π_m^k.

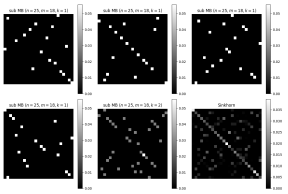
For some reasons that we will detail in the next section, for a fixed n and m we take k so that it is not higher than a certain value depending of n and m. We are showing the results in 3.



(a) OT matrix with $n = 25$, $m = 6$ for different values of k



(b) OT matrix with $n = 25$, $m = 12$ for different values of k



(c) OT matrix with $n = 25$, $m = 18$ for different values of k

Figure 3: OT matrices between distributions in 2D depending on the number of samples n and batch size m

Obviously, the result is better when m and k. However, choosing a higher m prevents choosing a higher k from 1 and therefore we observe high variance when m is high because it limits the value of k and so some data points are chosen too little. On the other hand, if we choose a lower value for m, which allows us to choose higher values for k, we also observe high variance due to the approximation error of the transport plan when we sample too few data points. Therefore, the better solution has to be chosen with this trade-off in mind.

2.3 Basic properties

We now state some basic properties on minibatch Wasserstein, that we will join with numerical examples if necessary.

The first proposition states an upper bound for k if we want to be worth in time computation.

Proposition 1 (Upper bound of k)**.** *Let n, $m \leq n$.*

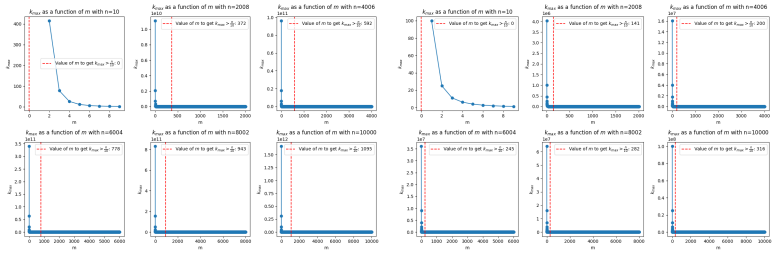
The computation of minibatch Wasserstein with Wasserstein distance is less expensive that the computation of the ground truth if:

$$k \leq \left(\frac{n}{m}\right)^3 \frac{\log(n)}{\log(m)} \tag{9}$$

The computation of minibatch Wasserstein with entropy loss or Sinkhorn divergence is less expensive that the computation of the ground truth if:

$$k \leq \left(\frac{n}{m}\right)^2 \tag{10}$$

This statement simply follows the time complexity of the Wasserstein distance with convex optimisation, which is in $\mathcal{O}(n^3 \log(n))$, and that of the Sinkhorn divergence, which is in $\mathcal{O}(n^2)$. Ideally, for time computation, we want to take the lowest value for k, but to maintain some coherence and a good approximation of $U_m(\alpha_n, \beta_n)$, we need to take the largest k possible. This gives us an extreme upper bound that we have to respect if we want minibatch theory to be computationally worthwhile. We show the evolution of k for different n when varying m in 5a, 5b for when we are dealing with Waserstein distance and Sinkhorn divergence.



(a) For Wasserstein (b) For Sinkhorn

Figure 4: Evolution of k_{lim} with respect to m for different values of n

We also take the liberty of plotting the limit of m when we want to sample a number proportional to n. We observe that, proportional to m, the value of k_max decreases as n increases. We also observe that k_{max} is higher with Wasserstein distance than with Sinkhorn divergence for small n, but when n grows Sinkhorn divergence allows a larger value for k than Wasserstein distance.

The second proposition concerns the optimal transport plan.

Proposition 2 (Admissible transportation plan)**.** *The transportation plan $\Pi_m(\alpha_n, \beta_n)$ is an admissible transportation plan between the full input distributions α_n, β_n and we have: $U_h(\alpha_n, \beta_n) \leq W(\alpha_n, \beta_n)$.*

The proof of this statement is available in [3]. Note that the fact that $\Pi_m(\alpha_n, \beta_n)$ is a valid transport plan does not mean that its subset $\Pi_m^k(\alpha_n, \beta_n)$ is. In general it is not for finite k.

It is interesting to know how many samples we need to be sure that all points have been sampled at least once.

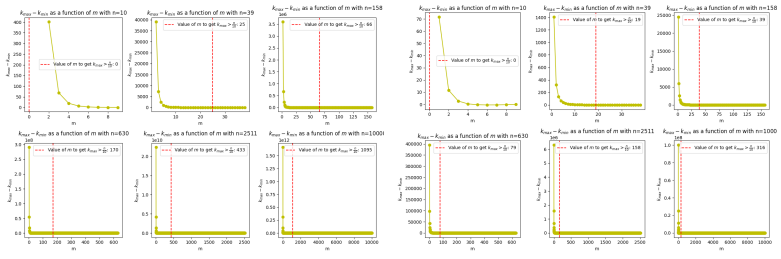
Proposition 3 (Probability of sampling)**.** *For a set of n points, the probability of sampling all the points at least one time when we are sampling k times m points uniformly is (noting A_i the number of different points that have been sampled at the i-th iteration)*

$$\mathbb{P}(A_k \geq n) = 1 - (1 - \frac{m}{n})^k \tag{11}$$

Hence with the probability at least $1 - \delta$, we sample all the points when $k \geq \frac{\log(\delta)}{\log(1-\frac{m}{n})}$

Proof: At each iteration, the probability that a particular point is not sampled is $(1 - 1/n) * (1 - 1/n - 1) * ... * (1 - 1/n - m + 1)$. Using the independence of each draw and the complementary, we get the telescopic product $\mathbb{P}(A_k \geq n) = 1 - \prod_{i=0}^{m-1}(1 - \frac{1}{n-i})^k = 1 - \prod_{i=0}^{m-1}(\frac{n-(i+1)}{n-i})^k$.

Using a confidence probability of 95% ($\delta = 0.05$), we have a formula of the minimum k to chose to sample each point. We can compare it to the maximum k computed in 4 to see admissible values for minibatch size m for different value of n. We show the result in 5.



(a) For Wasserstein (b) For Sinkhorn

Figure 5: Value of $k_{max} - k_{min}$ with respect to m for different values of n

We see that k_min is always below k_max, which shows that every value of m is admissible. This is a great thing since it justifies again to use the minibatch strategy.

An important question of the empirical estimator is whether it is biased. The following proposition state that it is.

Proposition 4 (Unbiased estimator)**.** *The transportation plan $\Pi_m(\alpha_n, \beta_n)$ is an admissible transportation plan between the full input distributions α_n, β_n and we have: $U_h(\alpha_n, \beta_n) \leq W(\alpha_n, \beta_n)$.*

The proof is in [3]. In spite of these appealing properties, using minibatch Wasserstein as a loss function presents the default that it is not a metric.

Proposition 5 (Positive and symmetry)**.** *The minibatch Wasserstein losses are positive and symmetric losses. However, they are not metrics since* $U_h(\alpha, \alpha) > 0$.

The proof is in [3].

[3] have also established asymptotic convergence for the estimator.

2.4 Asymptotic convergence

In this section, we present the asymptotic results established in [3]

Theorem 1 (Maximal deviation bound)**.** *Let* $\delta \in (0, 1))$, $k > 1$ *and* m *be fixed, and consider two distributions* α, β, *with bounded support. We have a deviation bound between* $U_h^k(\alpha_n, \beta_n)$ *and* $U_h(\alpha_n, \beta_n)$ *depending on the number of empirical data* n *and the number of batches* k, *with probability at least* $1 - \delta$ *on the draw of* α_n, β_n *and* D_k *we have:*

$$|U_h^k(\alpha_n, \beta_n) - U_h(\alpha, \beta)| \leq M_h(\sqrt{\frac{\log(\frac{2}{\delta})}{2Ent\frac{n}{m}}}) + \sqrt{\frac{2\log(\frac{2}{\delta})}{k}}) \tag{12}$$

where M_h *depends on* h *and scales at most as* $\mathcal{O}(\log(m))$.

The proof is given in [3] and he also gives us a deviation bound between $U_h^k(\alpha_n, \beta_n)$ and $U_h(\alpha_n, \beta_n)$.

This deviation bound shows that if we increase the number of data n and batches k while keeping the minibatch size m fixed, we get closer to the expectation. We will investigate the dependence on k and m in different scenarios in the numerical experiments. Remarkably, the bound does not depend on the dimension d of the space, which is an attractive property when optimising in high dimension.

The second quantity used to study the asymptotic behaviour is the minibatch OT matrix Pi_m. Since it is virtually impossible to compute in practice, we investigate the error on the boundary condition of Pi_m^k. In the following, we denote $\Pi_k(\alpha_n, \beta_n)_{(i)}$ by is the ith row of the matrix Π and $1 \in R^n$ is the vector full of 1.

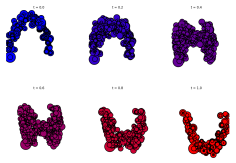
Theorem 2 (Distance to marginals)**.** *Let* $\delta \in (0, 1))$, $k > 1$ *and* m *be fixed, and consider two distributions* α, β. *For all* $k \geq 1$ *and* $1 \leq i \leq n$, *with probability at least* $1 - \delta$ *on the draw of* α, β *and* D_k *we have:*

$$|\Pi_k(\alpha_n, \beta_n)_{(i)} 1 - \frac{1}{n}| \leq \sqrt{\frac{2\log(2/\delta)}{k}} \tag{13}$$

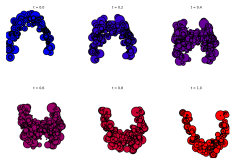
The proof is given in [3]. This theorem means that for high value of k, we are right to think that $\Pi_k(\alpha_n, \beta_n)$ becomes an admissible transport plan.

2.5 Minibatch Wasserstein Interpolation

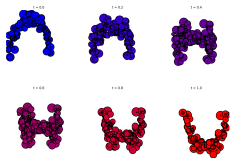
In this section we will interpolate between two sets of points. We know that the Wasserstein distance is a very natural geodesic distance, so we might be interested to see if the minibatch Wasserstein preserves this property. For this reason we have performed interpolations in different cases (the 2d and the 3d case), we show the results in 6 and 7.



(a) Interpolation with $m = 25$, $k = 16$

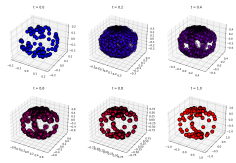


(b) Interpolation with $m = 50$, $k = 4$

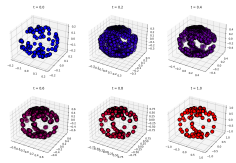


(c) Interpolation with $m = 75$, $k = 1$

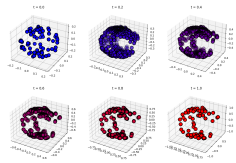
Figure 6: Minibatch Wasserstein Interpolation in 2D depending on the batch size m and the corresponding k



(a) Interpolation with $m = 25$, $k = 16$



(b) Interpolation with $m = 50$, $k = 4$



(c) Interpolation with $m = 75$, $k = 1$

Figure 7: Minibatch Wasserstein Interpolation in 3D depending on the batch size m and the corresponding k

We observe a very natural way of going from the first set to the second. Since everything is controlled by the calculation of the transport plan matrix, it's natural that each sample is sampled at least once. With 5 we can see that this condition is fulfilled with a high probability for $m = 25$ and $m = 50$, but not for $m = 75$, since it's associated with $k = 1$. Thus, the result with $m = 50$ and $k = 4$ seems to be very good, since the transport plan contains enough information to transport all points (k is high enough), and the transport is close to the optimal transport (m is high enough). Conversely, with $m = 75$ we see (net in the 3d case) that we lose the transport of some points, and with $m = 25$ we get a very natural way to transport α to β, but a little far from the optimal one. However, if we consider the case where every point is sampled, it seems to be better to use

2.6 Gradient and Optimisation

In this section, we investigate the optimisation properties of minibatch optimal transport (OT) losses to ensure the convergence of our loss functions within modern optimisation frameworks. We examine a typical parametric data fitting problem where we aim to fit a parametric model, denoted as $\lambda \to \beta(\lambda)$, to an unknown distribution α using the minibatch Wasserstein distance within a set λ in Euclidean space.

$$\min_{\lambda \in \Lambda} U_h(\alpha_n, \beta_\lambda) \tag{9}$$

Such problems are considered semi-discrete OT problems because one distribution is continuous while the other is discrete. For example, generative models, also known as minimal Wasserstein estimation, fall into this category [5]. The authors observed a successful application of stochastic gradient descent (SGD) in practice, yielding meaningful results with minibatches.

However, the empirical Wasserstein distance, which is known to be a biased estimator as discussed in [1], can lead to biased gradients, potentially causing SGD to fail. Our goal in this section is to show that, unlike the full Wasserstein distance, the minibatch strategy does not suffer from biased gradients.

However, since the original Wasserstein distance is not differentiable, we will primarily consider the entropic loss and the Sinkhorn divergence, which are differentiable. The entropic loss is not a measure, since $W_\epsilon(\alpha, \alpha) \neq 0$. The debiased sinkhorn corrects this and allows to get good behaviour, as we will show in the next section.

The following theorem, established in [3], states that the gradient of the minibatch is unbiased and thus justifies the use of SGD.

Theorem 3 (Exchange of Gradient and Expectation)**.** *Let $\lambda \in V$, where V is a nontrivial open set in $\mathbb{R}^p$. Assume α and ζ are compactly supported distributions. Let $X \sim \alpha^{\otimes m}$ and $Z \sim \zeta^{\otimes m}$ be two random variables in $\mathbb{R}^{m \times d}$. Suppose $\psi_\lambda : Z \to Y$ is differentiable with bounded gradients, and the ground cost C is C^1. Then, for the entropic loss and the Sinkhorn divergence:*

$$\nabla_\lambda \int_{X^{\otimes m}} \int_{Z^{\otimes m}} h(X, \psi_\lambda(Z))\, d\alpha^{\otimes m}(X)\, d\zeta^{\otimes m}(Z) =$$

$$\int_{X^{\otimes m}} \int_{Z^{\otimes m}} \nabla_\lambda h(X, \psi_\lambda(Z))\, d\alpha^{\otimes m}(X)\, d\zeta^{\otimes m}(Z)$$

The proof can be found in [3].

2.7 Minibatch Wasserstein Flow

We can now use this gradient property of minibatch Wasserstein with some examples. In particular, we explore the behaviour of the gradient with respect to the batch size m and the number of samples k.

Thus, in this section, we attempt to perform a Minibatch Wasserstein flow for matching between two sets of points by minimising the energy function:

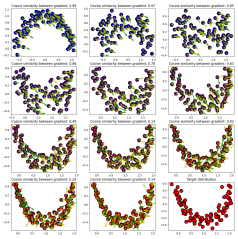
$$\varepsilon(z) = S_\epsilon(\frac{1}{n}\sum_i \delta_{z_i}, \frac{1}{m}\sum_i \delta_{y_i})$$

where S_ϵ is the debiased Sinkhorn score.

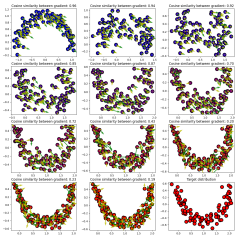
To compute the gradient, we chose to use backpropagation with Pytorch. Furthermore, we also compute the debiased Sinkhorn divergence on all samples to see what is different with the minibatch strategy.

Therefore, we decided to do this with dimension 2 and 3 to visualise the behaviour of the minibatch gradient and compare it with the true gradient (computed on the whole dataset).

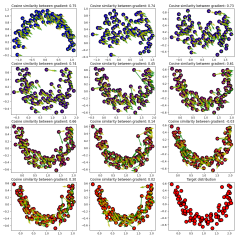
We show the 2d case in 8 and the 3d case in 9.



(a) Minibatch Wasserstein Flow with $m = 25$, $k = 16$



(b) Minibatch Wasserstein Flow with $m = 50$, $k = 4$

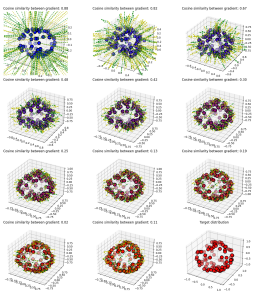


(c) Minibatch Wasserstein Flow with $m = 75$, $k = 1$

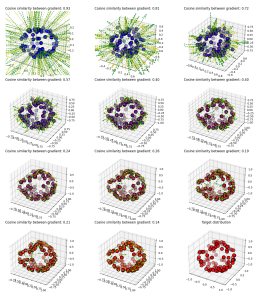
Figure 8: OT matrices between distributions in 2D depending on the number of samples n and batch size m

For the 2d case we decide to match a moon set of $n = 100$ with an inverted moon set of the same number of points. We show in 8 the influence of m and k when we vary m and take the limit of k given by 1. And for the 3d case we try to shift the surface of a sphere to the surface of a larger sphere. We decide to compare the gradient computed with minibatch with the true gradient. We plotted the two with the cosine similarity between them.

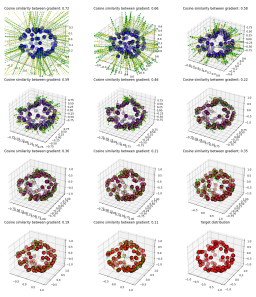
We see that the gradient is really good at the beginning and gets worse at the end. The reason for this is that minibatch works really well when we



(a) Minibatch Wasserstein Flow with $m = 25$, $k = 16$



(b) Minibatch Wasserstein Flow with $m = 50$, $k = 4$



(c) Minibatch Wasserstein Flow with $m = 75$, $k = 1$

Figure 9: OT matrices between distributions in 3D depending on the number of samples n and batch size m

want to connect 2 sets of points that are very far apart. So at the beginning, when we just need a rough direction, the minibatch plan will give us that. In the end, this is not what we get. This is also understandable if you consider that the two sets are initially very far apart and the sets are quite dense. I mean, looking at one point of one set or its neighbour doesn't have much effect on the direction of the transport. However, when the two sets are mixed, the correct direction has to be calculated with all the points. In fact, with the cosine similarity between gradients, we see that it is better to start with a high k (implying a low m) and end with a high m (implying a low k). Therefore, a solution to explore is to increase m with the end of the displacement (or training in a machine learning problem). In fact, m acts like a regularisation term, the closer we are to the end, the more we need to be more accurate and take $m \simeq n$.

3 Conclusion and perspective

In this report we recall and study results made in [3] on the use of minibatch strategy when we are dealing with OT. Therefore, we have recalled their results on the basic properties and asymptotic behaviour of the estimator. We saw a necessary trade-off between batch size m and number of draws k. We also see an upper bound on the number of draws when we fix the batch size to be computationally efficient. The higher the batch size m, the better the transport. Taking a higher value of the number of draws k implies that we have to consider the information coming from all points, and then we don't leave some points aside. A future work could be to elaborate a more complex strategy, which could consist in varying m and k along a gradient descent, as it already exists in machine learning with varying the learning rate during training. We also didn't study the unbiased version of

the minibatch Wasserstein, as proposed in the conclusion in [3].

4 Connection with the course

As well as enabling me to understand the paper and the concepts discussed in [3], the course and the numerical tours made me understand the influence of the regularisation parameter ϵ when we use the Sinkhorn algorithm. I was therefore able to choose the appropriate ϵ for each task. As a result, I have a better understanding of the role of the minibatch size m as a regularisation parameter. In practice, I first wanted to calculate all the util functions of OT by hand, as we can do in the numerical tours. And finally I preferred to use these functions, which are available on the POT Python package. All the drawing functions were strongly inspired by those presented in the numerical tours.

In order to compute the exact Wasserstein minibatch in 1D by sorting the sets, I wanted to compute in 2D with Gaussian distributions, since we know the optimal transport plan of it. However, I did not manage to compute the minibatch Wasserstein of it, mainly due to a misunderstanding of the continuous loss defined with minibatch.

References

[1] Marc G. Bellemare, Ivo Danihelka, Will Dabney, Shakir Mohamed, Balaji Lakshminarayanan, Stephan Hoyer, and Rémi Munos. The cramer distance as a solution to biased wasserstein gradients, 2017.

[2] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transportation distances, 2013.

[3] Kilian Fatras, Younes Zine, Rémi Flamary, Rémi Gribonval, and Nicolas Courty. Learning with minibatch wasserstein : asymptotic and gradient properties, 2021.

[4] Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trouve, and Gabriel Peyré. Interpolating between optimal transport and mmd using sinkhorn divergences. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 2681–2690. PMLR, 16–18 Apr 2019.

[5] Aude Genevay, Gabriel Peyre, and Marco Cuturi. Learning generative models with sinkhorn divergences. In Amos Storkey and Fernando Perez-Cruz, editors, *Proceedings of the Twenty-First International Conference*

on Artificial Intelligence and Statistics, volume 84 of *Proceedings of Machine Learning Research*, pages 1608–1617. PMLR, 09–11 Apr 2018.

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