Playing Around with the Numerical Tours

Mathias Vigouroux[*]

April 5, 2024

Abstract

The aim of this report is to display some experimentations with new data on the numerical tours, specifically for Optimal Transport. The aim is to play with the algorithmns to better understand what is happing.

1 Optimal Transport with Linear Programming

My code for this first numerical can be found on here.

1.1 Optimal Transport of Discrete Distributions

To start softly, we consider two discrete distributions

$$\alpha = \sum_{i=1}^{n} a_i \delta_{x_i} \quad \text{and} \quad \beta = \sum_{j=1}^{m} b_j \delta_{y_j}.$$

where n and m are the number of points, δ_x is the Dirac delta function at location $x \in \mathbb{R}^d$, and $(x_i)_i$ and $(y_j)_j$ are the positions of the Diracs in $\mathbb{R}^d$.

For this subsection, we will use $n = 60$ and $m = 80$. The initial data points of the numerics are α an unimodal Gaussian distribution and β a trivariate Gaussian distribution.

The data points are depicted in 1 with the size of each dot proportional to its probability density weight (1).

Let's compute the cost matrix defined as $C_{i,j} := \|x_i - x_j\|^2$.

Then, let's define the set of discrete couplings between α and β

$$U(a,b) := P \in_{+}^{n\times m} \forall i, \sum_j P_{i,j} = a_i,\ \forall j, \sum_i P_{i,j} = b_j.$$

The Kantorovitch formulation of the optimal transport reads

$$P^\star \in \arg \min_{P \in U(a,b)} \sum_{i,j} P_{i,j} C_{i,j}.$$

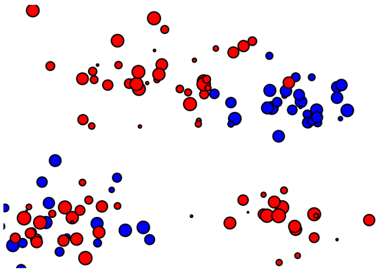

Figure 1: Data distribution

It is also important to check that the number of non-zero entries in $P^\star$ is $n+m-1$. Which is true since we find that : `Number of non-zero:  139 (n + m-1 = 139)`

The solution of the coupling can be displayed in (2), from which it is possible to display the connection defined by the optimal coupling in (3).

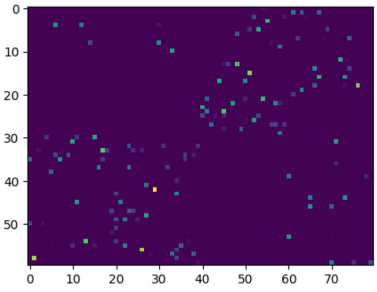

Figure 2: Coupling matirx

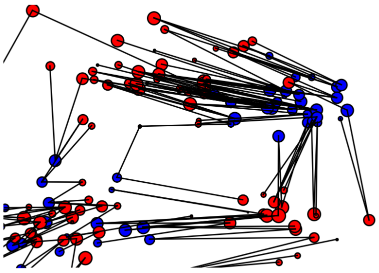

Figure 3: connections of the optimal coupling

[*] MVA, ENS-PLS: mathias.vigouroux@ens.psl.eu

1.2 Displacement Interpolation

One interesting things is that it is possible to move continuously from one distibution to the other. More precisely, for any $t \in [0,1]$, one can define a distribution μ_t such that $t \mapsto \mu_t$ defines a geodesic for the Wasserstein metric.

Since the W_2 distance is a geodesic distance, this geodesic path solves the following variational problem

$$\mu_t = \arg\min_{\mu} (1-t) W_2(\alpha, \mu)^2 + t W_2(\beta, \mu)^2.$$

This can be understood as a generalization of the usual Euclidean barycenter to the barycenter of distributions. Indeed, in the case that $\alpha = \delta_x$ and $\beta = \delta_y$, one has $\mu_t = \delta_{x_t}$ where $x_t = (1-t)x + ty$.

Once the optimal coupling $P^\star$ has been computed, the interpolated distribution is obtained as

$$\mu_t = \sum_{i,j} P^\star_{i,j} \delta_{(1-t)x_i + t y_j}.$$

Display the evolution of μ_t for a varying value of $t \in [0,1]$ in 4.

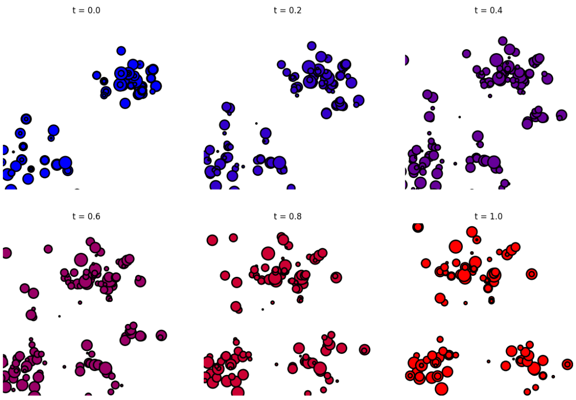

Figure 4: Geodesic Evolution

1.3 Optimal Assignement

In the case where $n = m$ and the weights are uniform $a_i = 1/n, b_j = 1/n$, one can show that there is at least one optimal transport coupling which is actually a permutation matrix. This properties comes from the fact that the extremal point of the polytope $U(1,1)$ are permutation matrices.

This means that there exists an optimal permutation $\sigma^\star \in \Sigma_n$ such that

$$P^\star_{i,j} = \begin{cases} 1 & \text{if } j = \sigma^\star(i), \\ 0 & \text{otherwise.} \end{cases}$$

where Σ_n is the set of permutation (bijections) of $\{1, \ldots, n\}$.

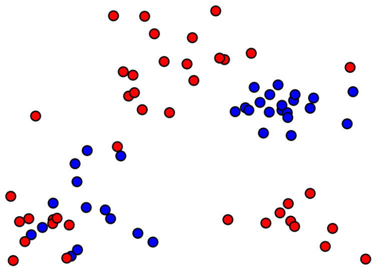

Figure 5: Data point clouds x and y

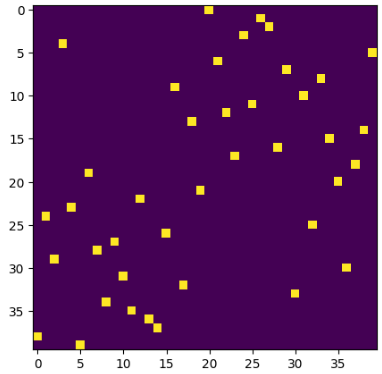

Figure 6: Binary coupling

This permutation thus solves the so-called optimal assignement problem

$$\sigma^\star \in arg \min_{\sigma \in \Sigma_n} \sum_i C_{i,\sigma(j)}.$$

It is possible to display the new data point clouds5:

and to Show that P 6 is a binary permutation matrix because in the end it is possible to see that one point is assigned to one other point 7.

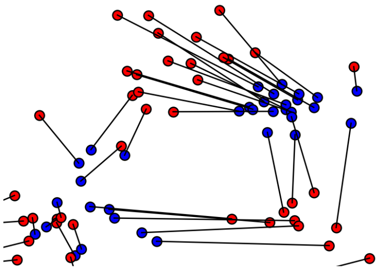

Figure 7: Connection for a binary coupling

2 Entropic Regularization of Optimal Transport

My code for this second numerical can be found on here.

2.1 Entropic Regularization of Optimal Transport

This second section exposes the general methodology of regularizing the optimal transport (OT) linear program using entropy. This allows to derive fast computation algorithm based on iterative projections according to a Kulback-Leiber divergence.

We consider two input histograms $a, b \in \Sigma_n$, where we denote the simplex in $\mathbb{R}^n$

$$\Sigma_n := \left\{ a \in \mathbb{R}^n_+, \sum_i a_i = 1 \right\}.$$

We consider the following discrete regularized transport

$$W_\epsilon(a,b) := \min_{P \in U(a,b)} < C, P > -\epsilon E(P).$$

where the polytope of coupling is defined as

$$U(a,b) := \left\{ P \in (\mathbb{R}^+)^{n \times m}, P\mathbb{1}_m = a, P^T\mathbb{1}_n = b \right\},$$

where $\mathbb{1}_n := (1, \ldots, 1)^T \in \mathbb{R}^n$, and for $P \in \mathbb{R}^{n \times m}_+$, we define its entropy as

$$E(P) := -\sum_{i,j} P_{i,j}(\log(P_{i,j}) - 1).$$

When $\epsilon = 0$ one recovers the classical (discrete) optimal transport.

Here the matrix $C \in (\mathbb{R}^+)^{n \times m}$ defines the ground cost, i.e. $C_{i,j}$ is the cost of moving mass from a bin indexed by i to a bin indexed by j.

The regularized transportation problem can be re-written as a projection

$$W_\epsilon(a,b) = \epsilon \min_{P \in U(a,b)} KL(P|K)$$

where

$$K_{i,j} := e^{-\frac{C_{i,j}}{\epsilon}}$$

of the Gibbs kernel K according to the Kullback-Leibler divergence.

The Kullback-Leibler divergence between $P, K \in \mathbb{R}^{n \times m}_+$ is

$$KL(P|K) := \sum_{i,j} P_{i,j}(\log(\frac{P_{i,j}}{K_{i,j}}) - 1).$$

Given a convex set $\mathcal{C} \subset \mathbb{R}^N$, the projection according to the Kullback-Leiber divergence is defined as

$$Proj^{KL}_{\mathcal{C}}(\xi) = arg \min_{\pi \in \mathcal{C}} KL(\pi|\xi).$$

2.2 Iterative Bregman Projection Algorithm

Given affine constraint sets $(\mathcal{C}_1, \mathrm{C}_2)$, we aim at computing $Proj^{KL}_{\mathcal{C}}(K)$ where $\mathcal{C} = \mathcal{C}_1 \cap \mathcal{C}_2$ (this description can of course be extended to more than 2 sets).

This can be achieved, starting by $P_0 = K$, by iterating $\forall \ell \geq 0$, $P_{2\ell+1} = Proj^{KL}_{\mathcal{C}_\infty}(P_{2\ell})$ and $P_{2\ell+2} = Proj^{KL}_{\mathcal{C}_\in}(P_{2\ell+1})$.

One can indeed show that $P_\ell \to Proj^{KL}_{\mathcal{C}}(K)$.

2.3 Sinkhorn's Algorithm

A fundamental remark is that the optimality condition of the entropic regularized problem shows that the optimal coupling P_ϵ necessarily has the form

$$P_\epsilon = DiaguKDiagv$$

where the Gibbs kernel is defined as

$$K := e^{-\frac{C}{\epsilon}}.$$

One thus needs to find two positive scaling vectors $u \in \mathbb{R}^n_+$ and $v \in \mathbb{R}^m_+$ such that the two following equality holds $P\mathbb{1} = u \odot (Kv) = a$ and$P^\top\mathbb{1} = v \odot (K^T u) = b$.

Sinkhorn's algorithm alternate between the resolution of these two equations, and reads $u \longleftarrow \frac{a}{Kv}andv \longleftarrow \frac{b}{K^\top u}$.

2.4 Transport between point clouds

We first test the method for two input measures that are uniform measures (i.e. constant histograms) supported on two point clouds (that do not necessarily have the same size).

We thus first load two points clouds $x = (x_i)_{i=1}^n, y = (y_i)_{i=1}^m$, where $x_i, y_i \in \mathbb{R}^2$. For each cloud the number of points is, $N = (n, m)$. We will use $N = (250, 300)$.

The type of data points drawn will be from two squares for x, and from an anulus for y 8.

It is then possible to implement the Sinkhorn algorithm. Display the evolution of the constraints satisfaction errors $||P\mathbb{1} - a||_1$ and $||P^\top\mathbb{1} - b||$.

After 10000 it is possible to implement the logarithmic decay of the satisfaction errors in a logplot

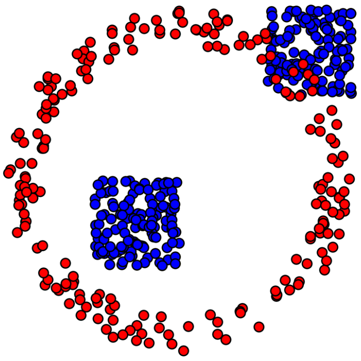

Figure 8: The two point clouds

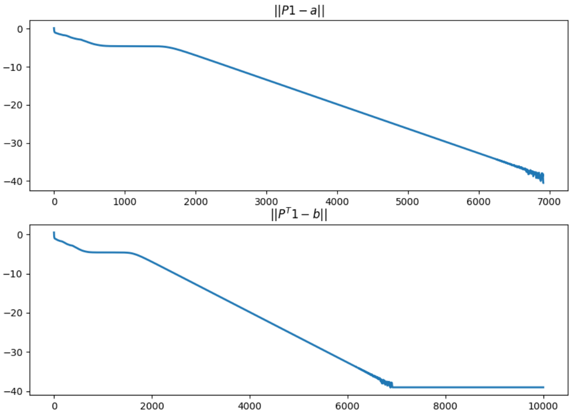

Figure 9: Logplot of the constrain error evolution

Depending on the ϵ the coupling can vary, ressembling more and more the optimal coupling. However, **if ϵ is too small the coupling fails to work**. In my case, I found that $\epsilon \geq 0.012$.

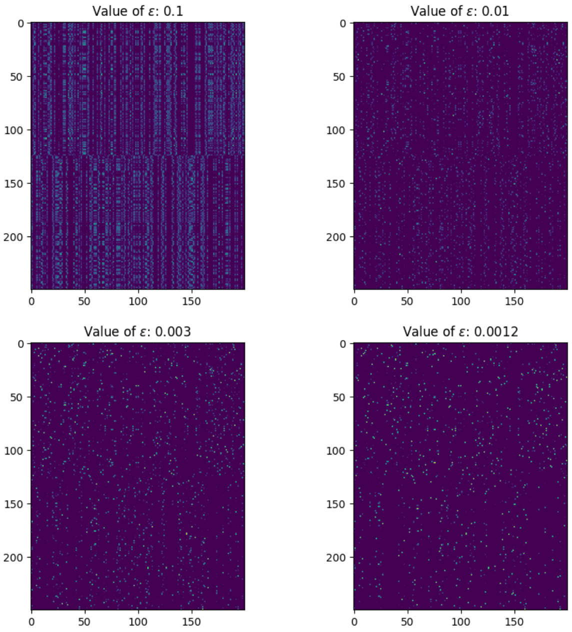

Figure 10: Logplot of the constrain error evolution

Finally, it is possible to represent this coupling.

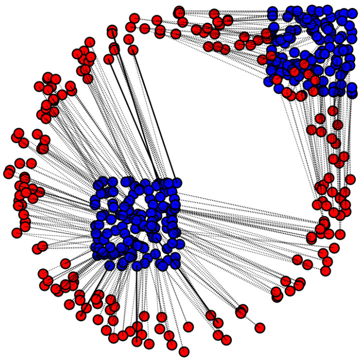

Figure 11: Coupling of two squares with an anulus

3 Transport Between Histograms

We now consider a different setup, where the histogram values a, b are not uniform, but the measures are defined on a uniform grid $x_i = y_i = i/n$. They are thus often referred to as "histograms. For my dataset, I decided to use both the mean and standard deviation different for the two distributions 12.

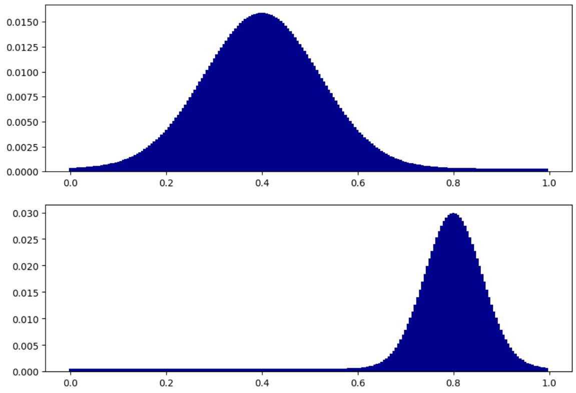

Figure 12: Two histograms

From which it is possible to once again plot the decay of the error 13.

One can compute an approximation of the transport plan between the two measure by computing the so-called barycentric projection map

$$t_i \in [0,1] \longmapsto s_j := \frac{\sum_j P_{i,j} t_j}{\sum_j P_{i,j}} = \frac{[u \odot K(v \odot t)]_j}{a_i}.$$

where $\odot$ and $\frac{\cdot}{\cdot}$ are the entry-wise multiplication and division.

This computation can thus be done using only multiplication with the kernel K.

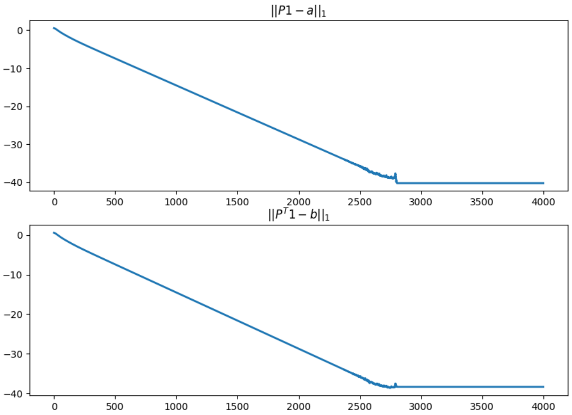

Figure 13: Logplot of the constrain error evolution of two histograms

From that it is possible to display the transport map, super-imposed over the coupling 14.

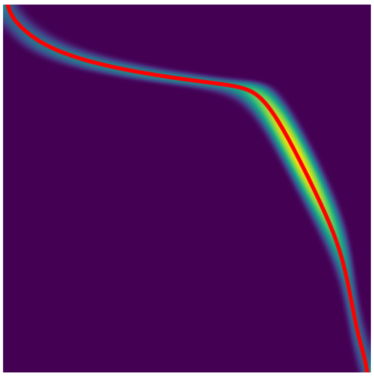

Figure 14: Transport map (in red) and the coupling of the two histograms (background)

3.1 Wasserstein Barycenters

Instead of computing transport, we now turn to the problem of computing barycenter of R input measures $(a_k)_{k=1}^R$. A barycenter b solves

$$\min_b \sum_{k=1}^{R} W_\gamma(a_k, b)$$

where $_k$ are positive weights with $\sum_k {}_k = 1$. This follows the definition of barycenters.

I decided to use 96 × 96 images that I created myself by scraping the net and then reshaping them 15.

In this specific case, the kernel K associated with the squared Euclidean norm is a convolution with a Gaussian filter

$$K_{i,j} = e^{-i/N - j/N^2/\epsilon}$$

where here (i, j) are 2-D indexes.

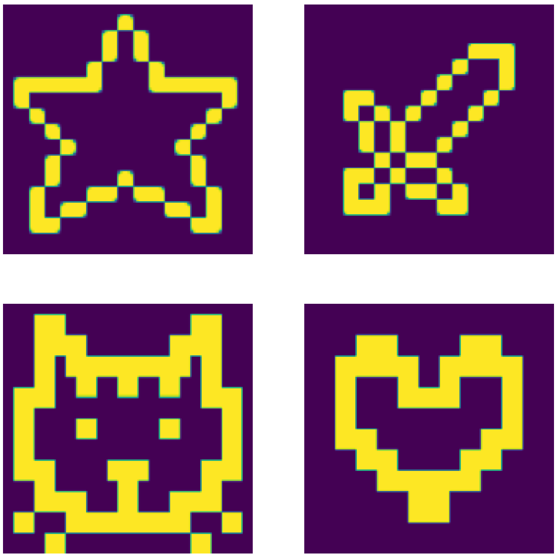

Figure 15: A star, a sword, a cat and a heart

The multiplication against the kernel, i.e. $K(a)$, can now be computed efficiently, using fast convolution methods.

Now we will define the kernel K. We use here the fact that the convolution is separable to implement it using only 1-D convolution, which further speeds up computations.

It is then possible to display the application of the K kernel on one of the input histogram 16.

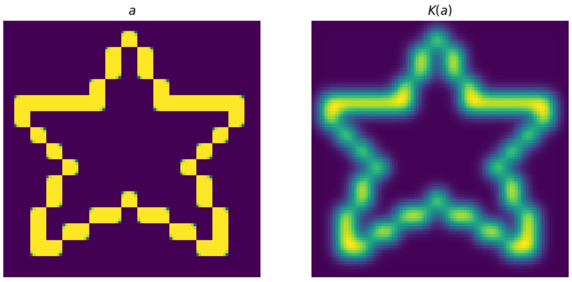

Figure 16: The kernel transformation of a star

The problem of Barycenter computation boilds down to optimizing over couplings $(P_k)_{k=1}^R$, and that this can be achieved using iterative a Sinkhorn-like algorithm, since the optimal coupling has the scaling form

$$P_k = diagu_k K diagv_k$$

for some unknown positive weights (u_k, v_k)

The first step of the Bregman projection method corresponds to the projection on the fixed marginals constraints $P^k \nVdash = a_k$. This is achieved by updating

$$\forall k = 1, \ldots, R, \quad u_k \longleftarrow \frac{a_k}{K(v_k)}.$$

The second step of the Bregman projection method corresponds to the projection on the equal marginals constraints $\forall k, P_k^{\top\mathbb{1}=b}$ for a common barycenter target b. This is achieved by first computing the target barycenter b using a geometric means

$$\log(b) := \sum_k \lambda_k \log(u_k \odot K(v_k)).$$

It is then possible to display b 17, the first iteration of the algorithm.

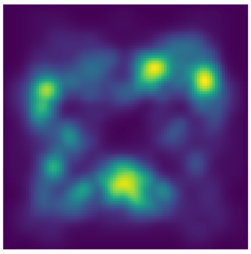

Figure 17: First iteration of Bregman algorithm

And then one can update the scaling by a Sinkhorn step using this newly computed histogram b as follow (note that $K = K^\top$ here):

$$\forall k = 1, \ldots, R, \quad v_k \longleftarrow \frac{b}{K(u_k)}.$$

From that it is possible to Implement the iterative algorithm to compute the iso-barycenter of the measures. Plot the decay of the error $\sum_k P_k \mathbb{1} - a_k$.

Then, it is possible to display the Wasserstein Barycenter 18.

What is quite interesting, is that human could stll interpret this image as the face of something even only one out of four images represent a face.

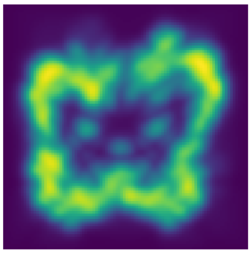

Figure 18: Barycenter

Finally it is possible, to have an insight at how varying the weight of an interpolation inside the square of the four pictures gives intermediate and interesting looking pictures 19.

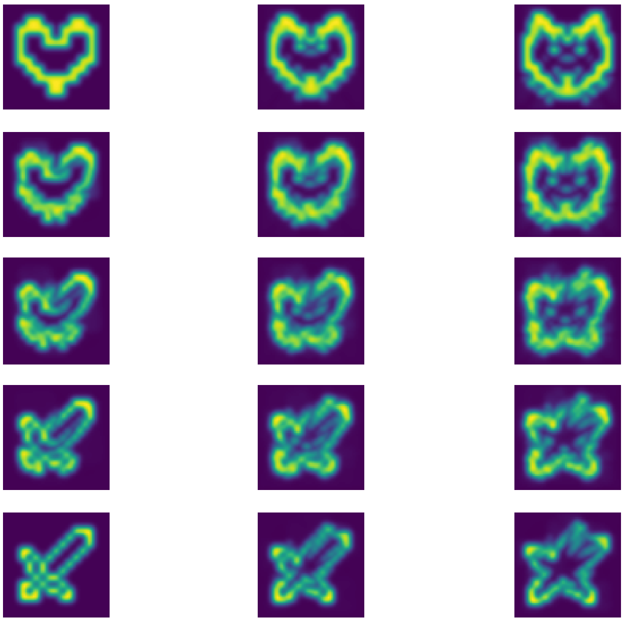

Figure 19: Moving between the four images gives rise to intersting looking creatures : for instance in between a cat an a star, an ant seems to appear due to the more angular shapes, while in the middle of the cat and the heart it is more looking like an elephant since the ears of the cat are becoming more round (like an elephant and horn seems to appear also).

4 Conclusion

In this document, we have considered first very easy implementation of Optimal transport problem and we slowly incread the complexity of the tasks finshing by using enthropic regularization of optimal transport to be able to shifht between different images once they were considered as histogramms.