A modern application of Linear Programming: Optimal transport – Part II

Numerics and applications

Disclaimers

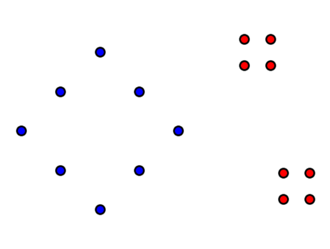
You will not be tested on that, and this will not help you to understand better Linear Programming.

Disclaimers

You will not be tested on that, and this will not help you to understand better Linear Programming.

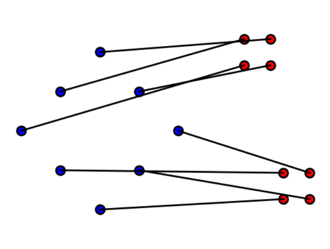
The goal is to present you a modern direction of research where Linear Programming plays a role.

Reminder: last lecture



Two clouds of points, in blue and red.

Reminder: last lecture



Two clouds of points, in blue and red.

By solving a LP, we get get:

- a matching between the two clouds of points (the optimal solution),
- and a distance measuring how close the two clouds are (the value).

1. Numerics with entropic regularization
2. More on shape interpolation
3. Color Transfer
4. Fluid dynamics

1. Numerics with entropic regularization

Minimizing smooth functions

Calculus

If $f : \mathbb{R}^n \to \mathbb{R}$ is a smooth function, to look for its minimum one can check the critical points. They are the solutions of $\nabla f(\mathbf{x}) = \mathbf{0}$, where

$$\nabla f = \begin{pmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \dots & \frac{\partial f}{\partial x_n} \end{pmatrix}^\top .$$

Minimizing smooth functions

Calculus

If $f : \mathbb{R}^n \to \mathbb{R}$ is a smooth function, to look for its minimum one can check the critical points. They are the solutions of $\nabla f(\mathbf{x}) = \mathbf{0}$, where

$$\nabla f = \begin{pmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \dots & \frac{\partial f}{\partial x_n} \end{pmatrix}^\top .$$

Why can't it work with Linear Programming?

Minimizing smooth functions

Calculus

If $f : \mathbb{R}^n \to \mathbb{R}$ is a smooth function, to look for its minimum one can check the critical points. They are the solutions of $\nabla f(\mathbf{x}) = \mathbf{0}$, where

$$\nabla f = \begin{pmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \dots & \frac{\partial f}{\partial x_n} \end{pmatrix}^\top .$$

Why can't it work with Linear Programming?

The objective function is $f(\mathbf{x}) = \mathbf{c}^\top \mathbf{x}$ and $\nabla f = \mathbf{c}$. The gradient never vanishes!

This is because in Linear Programming the variable $\mathbf{x}$ is constrained to be in the feasible region, and the constraints play a very important role.

Optimal transport problem

Data: weights a_i and b_j, transport cost C_{ij} with $i \in \{1, 2, \ldots, n\}$ and $j \in \{1, 2, \ldots, m\}$.

Optimal transport problem

We minimize among all P

$$\sum_{i=1}^{n} \sum_{j=1}^{m} P_{ij} C_{ij}$$

subject to

$$\begin{array}{rcll} \sum_{j=1}^{m} P_{ij} & = & a_i & \forall i \in \{1, 2, \ldots, n\} \\ \sum_{i=1}^{n} P_{ij} & = & b_j & \forall j \in \{1, 2, \ldots, m\} \end{array}$$

and the non negativity constraints $P_{ij} \geqslant 0$ for all i, j.

The complicated constraints to handle from a calculus point of view are the non negativity constraints.

Entropy

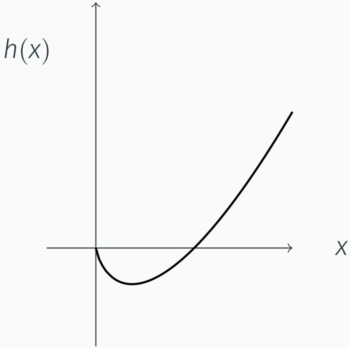
"You should call it entropy, for two reasons. In the first place your uncertainty function has been used in statistical mechanics under that name, so it already has a name. In the second place, and more important, **no one really knows what entropy really is, so in a debate you will always have the advantage**."

Von Neumann to Claude Shannon about entropy in the theory of information, my emphasis.

Entropy

In this lecture, entropy is the real valued function

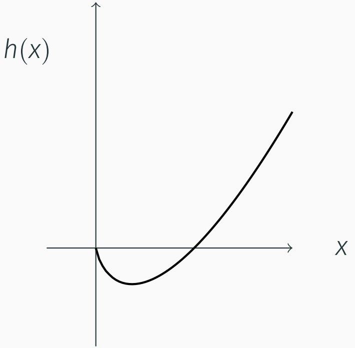
$$h : x \mapsto x \ln(x).$$



Entropy

In this lecture, entropy is the real valued function

$$h : x \mapsto x \ln(x).$$



Importantly, this function satisfies $h'(0) = -\infty$. To decrease h, it is always good to move x away from 0.

Entropic regularization of optimal transport [1]

Entropic regularized optimal transport problem

Same data as before, with an additional $\varepsilon > 0$. We minimize among all P

$$\sum_{i=1}^{n}\sum_{j=1}^{m} P_{ij}C_{ij} + \varepsilon \sum_{i=1}^{n}\sum_{j=1}^{m} \underbrace{P_{ij}\ln(P_{ij})}_{=h(P_{ij})}$$

subject to

$$\begin{array}{rcll} \sum_{j=1}^{m} P_{ij} & = & a_i & \forall i \in \{1,2,\dots,n\} \\ \sum_{i=1}^{n} P_{ij} & = & b_j & \forall j \in \{1,2,\dots,m\} \end{array}$$

and the non negativity constraints $P_{ij} \geqslant 0$ for all i, j.

[1] Cuturi. *Sinkhorn distances: Lightspeed computation of optimal transport*, 2013.

Entropic regularization of optimal transport [1]

Entropic regularized optimal transport problem

Same data as before, with an additional $\varepsilon > 0$. We minimize among all P

$$\sum_{i=1}^{n}\sum_{j=1}^{m} P_{ij}C_{ij} + \varepsilon \sum_{i=1}^{n}\sum_{j=1}^{m} \underbrace{P_{ij}\ln(P_{ij})}_{=h(P_{ij})}$$

subject to

$$\begin{array}{rcll} \sum_{j=1}^{m} P_{ij} & = & a_i & \forall i \in \{1,2,\ldots,n\} \\ \sum_{i=1}^{n} P_{ij} & = & b_j & \forall j \in \{1,2,\ldots,m\} \end{array}$$

and the non negativity constraints $P_{ij} \geqslant 0$ for all i, j.

If $\varepsilon > 0$ is small, the solution of this problem is closed to the original one.

[1] Cuturi. *Sinkhorn distances: Lightspeed computation of optimal transport*, 2013.

Entropic regularization of optimal transport [1]

Entropic regularized optimal transport problem

Same data as before, with an additional $\varepsilon > 0$. We minimize among all P

$$\sum_{i=1}^{n}\sum_{j=1}^{m} P_{ij}C_{ij} + \varepsilon \sum_{i=1}^{n}\sum_{j=1}^{m} \underbrace{P_{ij}\ln(P_{ij})}_{=h(P_{ij})}$$

subject to

$$\begin{array}{rcll} \sum_{j=1}^{m} P_{ij} & = & a_i & \forall i \in \{1,2,\dots,n\} \\ \sum_{i=1}^{n} P_{ij} & = & b_j & \forall j \in \{1,2,\dots,m\} \end{array}$$

and the non negativity constraints $P_{ij} \geqslant 0$ for all i, j.

If $\varepsilon > 0$ is small, the solution of this problem is closed to the original one.

Thanks $h'(0) = -\infty$, the minimizer will satisfy $P_{ij} > 0$ for all i, j: we can drop the non negativity constraint.

[1] Cuturi. *Sinkhorn distances: Lightspeed computation of optimal transport*, 2013.

Solution of the entropic regularized optimal transport

This is no longer a LP, but for this approximate problem we can use calculus to find critical points. In this case there is only one and it is a global minimizer.

Solution of the entropic regularized optimal transport

This is no longer a LP, but for this approximate problem we can use calculus to find critical points. In this case there is only one and it is a global minimizer.

We still have Lagrange multipliers (a.k.a dual variables) u_i and v_j for the equality constraints. The optimality conditions (the "$\nabla f = \mathbf{0}$") read

$$C_{ij} + \varepsilon \ln(P_{ij}) + \varepsilon = u_i + v_j \qquad \forall i, j.$$

Solution of the entropic regularized optimal transport

This is no longer a LP, but for this approximate problem we can use calculus to find critical points. In this case there is only one and it is a global minimizer.

We still have Lagrange multipliers (a.k.a dual variables) u_i and v_j for the equality constraints. The optimality conditions (the "$\nabla f = \mathbf{0}$") read

$$C_{ij} + \varepsilon \ln(P_{ij}) + \varepsilon = u_i + v_j \qquad \forall i, j.$$

Calling $\tilde{u}_i = \exp(\varepsilon^{-1} u_i - 1)$ and $\tilde{v}_j = \exp(\varepsilon^{-1} v_j)$, this reads

$$P_{ij} = \tilde{u}_i \exp\left(-\frac{C_{ij}}{\varepsilon}\right) \tilde{v}_j \qquad \forall i, j.$$

Solution of the entropic regularized optimal transport

This is no longer a LP, but for this approximate problem we can use calculus to find critical points. In this case there is only one and it is a global minimizer.

We still have Lagrange multipliers (a.k.a dual variables) u_i and v_j for the equality constraints. The optimality conditions (the "$\nabla f = \mathbf{0}$") read

$$C_{ij} + \varepsilon \ln(P_{ij}) + \varepsilon = u_i + v_j \qquad \forall i, j.$$

Calling $\tilde{u}_i = \exp(\varepsilon^{-1} u_i - 1)$ and $\tilde{v}_j = \exp(\varepsilon^{-1} v_j)$, this reads

$$P_{ij} = \tilde{u}_i \exp\left(-\frac{C_{ij}}{\varepsilon}\right) \tilde{v}_j \qquad \forall i, j.$$

Then, one can very efficiently find $\tilde{u}_i$ and $\tilde{v}_j$ in such a way that the equality constraints of P are satisfied via a procedure called *Iterative Proportional Fitting Procedure* with only matrix vector multiplications!

How fast is it?[2]

For clouds of points in dimension 2 or 3, with moderately small ε, with a multiscale approach, with parallel computations and on a reasonable computer,

[2]Figures taken from the Library GeomLoss: `https://www.kernel-operations.io/geomloss/`.

How fast is it?[2]

For clouds of points in dimension 2 or 3, with moderately small ε, with a multiscale approach, with parallel computations and on a reasonable computer, it can take less than a second for an entropic regularized optimal transport problem with $n = m = 10^6$.

This is an optimization problem with 10^{12} decision variables and 2×10^6 constraints!

[2] Figures taken from the Library GeomLoss: `https://www.kernel-operations.io/geomloss/`.

Take home message

Key points:

- By changing slightly the problem (adding $\varepsilon \times$ [entropy]), one moves the optimal solution in the *interior* of the polytope of the constraints, that is $P_{ij} > 0$ for all i, j. Then, calculus is available: this is the idea behind "interior point methods".

Take home message

Key points:

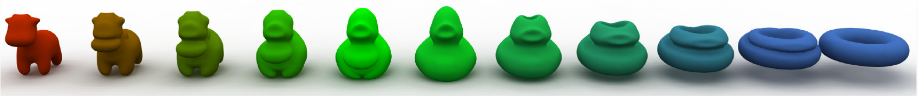
- By changing slightly the problem (adding $\varepsilon \times$ [entropy]), one moves the optimal solution in the *interior* of the polytope of the constraints, that is $P_{ij} > 0$ for all i, j. Then, calculus is available: this is the idea behind "interior point methods".
- In the optimal transport problem, the modification is done with this entropic regularization. The algebraic structure of the optimality conditions allows for a simple procedure to solve the problem.

2. More on shape interpolation

Reminder: last lecture

Once a matching is found, one can make the mass travel at constant speed to interpolate between point clouds approximating shapes.

Interpolation in three dimensions[3]



A cow being interpolated to a duck, and then to a doughnut.

[3] Solomon, de Goes, Peyré, Cuturi, Butscher, Nguyen, Du and Guibas. *Convolutional Wasserstein Distances: Efficient Optimal Transportation on Geometric Domains*, 2015.

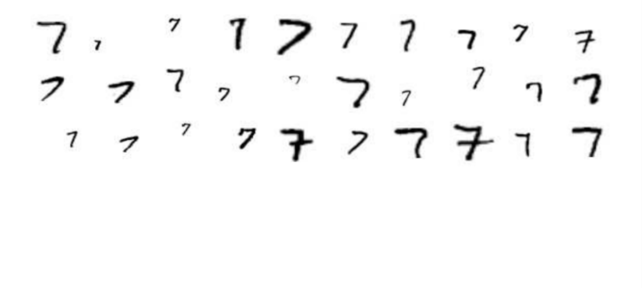
Interpolation over surfaces[4]

The mass (in blue) is moving over a fixed surface (here a hand). This was *not* generated with entropic regularization.

[4]Lavenant, Claici, Chien and Solomon. *Dynamical Optimal Transport on Discrete Surfaces*, 2018.

Barcyenters[5]

We can compute average between more than 2 shapes!

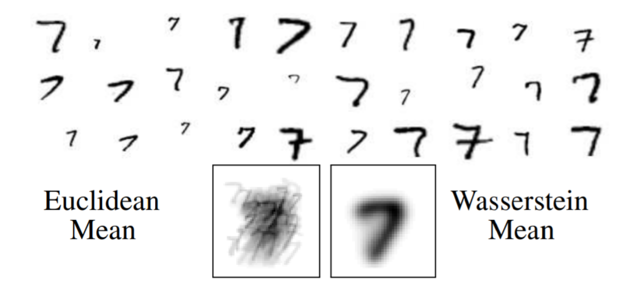


- Top: scans of handwritten digit 7.

[5] Uribe, Dvinskikh, Dvurechensky, Gasnikov and Nedić. *Distributed Computation of Wasserstein Barycenters over Networks*, 2018.

Barcyenters[5]

We can compute average between more than 2 shapes!



- Top: scans of handwritten digit 7.
- Bottom left: pixel per pixel mean of the digits.
- Bottom right: mean of the digits using regularized optimal transport to compare shapes.

[5] Uribe, Dvinskikh, Dvurechensky, Gasnikov and Nedić. *Distributed Computation of Wasserstein Barycenters over Networks*, 2018.

3. Color Transfer

Matching the colors of an image[6]

Image 1

Image 2

[6] Pictures generated by Peyré and published in Santambrogio, *Optimal transport for applied mathematicians*, 2015.

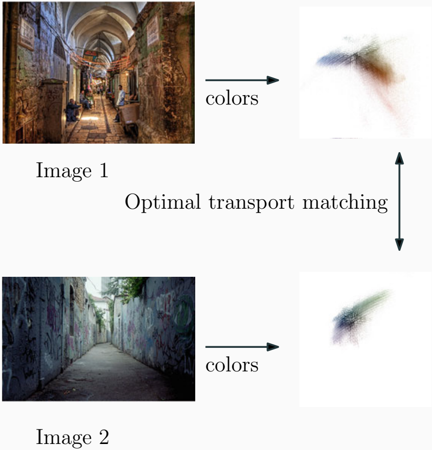
Matching the colors of an image[6]

Image 1

Image 2

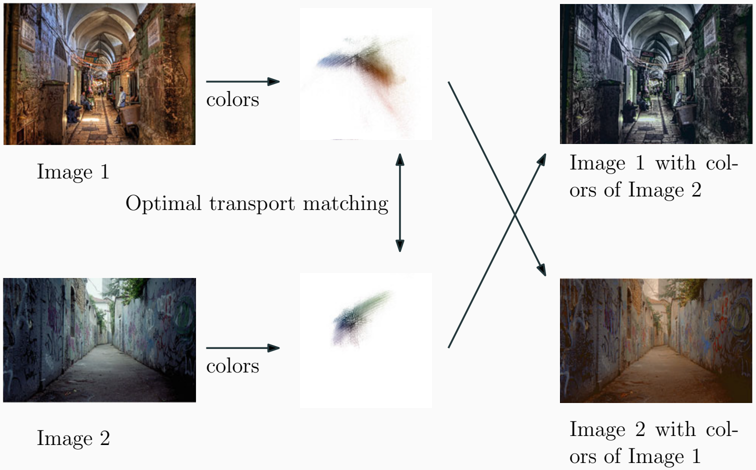
[6] Pictures generated by Peyré and published in Santambrogio, *Optimal transport for applied mathematicians*, 2015.

Matching the colors of an image[6]



[6] Pictures generated by Peyré and published in Santambrogio, *Optimal transport for applied mathematicians*, 2015.

Matching the colors of an image[6]



[6]Pictures generated by Peyré and published in Santambrogio, *Optimal transport for applied mathematicians*, 2015.

4. Fluid dynamics

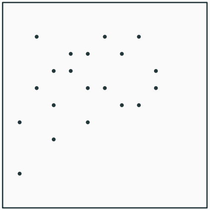
Optimal transport in computational fluid dynamics

One approach: representing your fluid by an assembly of particles (like “hard spheres”). The complicated part is to take in account the interaction between particles.

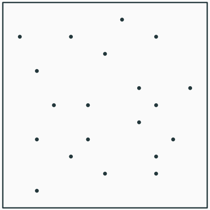
Optimal transport in computational fluid dynamics

One approach: representing your fluid by an assembly of particles (like "hard spheres"). The complicated part is to take in account the interaction between particles.

Incompressibility: the distribution of particles should be as uniform as possible



Not uniformly distributed

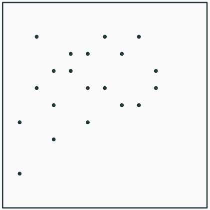


More uniformly distributed

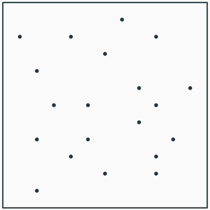
Optimal transport in computational fluid dynamics

One approach: representing your fluid by an assembly of particles (like "hard spheres"). The complicated part is to take in account the interaction between particles.

Incompressibility: the distribution of particles should be as uniform as possible



Not uniformly distributed

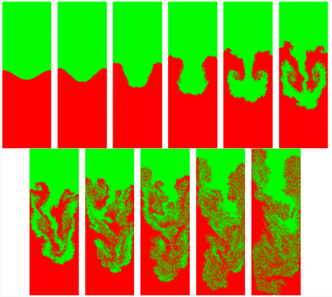


More uniformly distributed

One can use an optimal transport distance (that is the value of an optimal transport problem) to compute how "uniformly distributed" the set of particles is. This is one block of the computational fluid dynamics solver.

Examples[7] [8]

In these simulations, the optimal transport problem is not solved by entropic regularization.



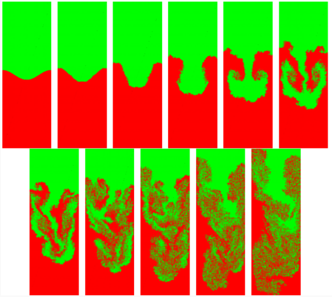
Two fluids: one denser on top, one lighter in the bottom. Total of 5×10^4 particles.

[7] Gallouët and Mérigot. *A Lagrangian Scheme à la Brenier for the Incompressible Euler Equations*, 2018.

[8] Lévy and Schwindt. *Notions of optimal transport theory and how to implement them on a computer*, 2018

Examples[7] [8]

In these simulations, the optimal transport problem is not solved by entropic regularization.



Similar instability, but in 3 dimensions: `https://www.youtube.com/watch?v=C1WQuwvMgD0`

Two fluids: one denser on top, one lighter in the bottom. Total of 5×10^4 particles.

[7] Gallouët and Mérigot. *A Lagrangian Scheme à la Brenier for the Incompressible Euler Equations*, 2018.

[8] Lévy and Schwindt. *Notions of optimal transport theory and how to implement them on a computer*, 2018