Geometric data analysis

Lecture 6/7 – Probability distributions

Jean Feydy

HeKA team, Inria Paris, Inserm, Université Paris-Cité

Thursday, 9am–12pm – 7 lectures

Faculté de médecine, Hôpital Cochin, rooms 2001 + 2005

Validation: project + quizz

Recap of the previous lectures

To mitigate the **curse of dimensionality**, we use:

- **Expert** knowledge: high-quality features.
- Relevant **families** of functions: kernels, convolutional networks.
- Relevant **neighborhood** structures: graphs.

Main challenge: **local** implementation $\Longrightarrow$ **global** understanding.

Produce **guidelines** and **insights** for practitioners.

A continuous vocabulary for data sciences [Fey20]

Lecture 5 – From discrete graphs to **continuous spaces**:

- The Poincaré disk.
- Local metrics and geodesics.

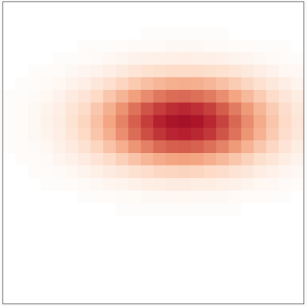
Lecture 6 – From discrete samples to **continuous distributions**:

- **Why do we care** about probability distributions?
- Information **geometry**, kernels and optimal transport.
- **Lab session** on gradient descent.

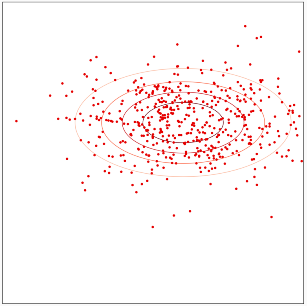
$\Longrightarrow$ Chapter 3 of my PhD thesis, *Geometric data analaysis, beyond convolutions*.

What is a probability distribution?

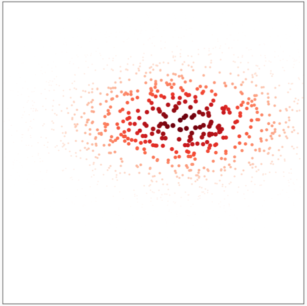
Probability distribution α = weights a_i at locations x_i



Histogram:
variable weights a_i,
fixed locations x_i.



Sample:
fixed weights $1/\mathrm{N}$,
variable locations x_i.

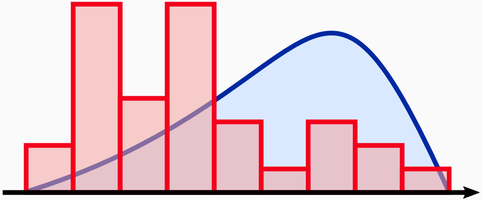


Weighted point cloud:
variable weights a_i,
variable locations x_i.

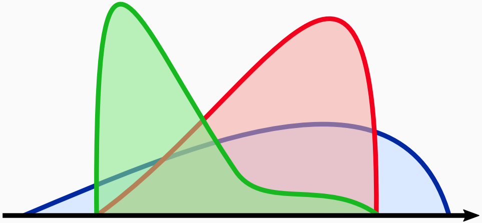
Discrete sum $\alpha = \sum_{i=1}^{\mathrm{N}} a_i \delta_{x_i} \quad \Longrightarrow \quad$ **Continuous** density $\alpha = \int_x a(x)\,\mathrm{d}x$.

Today, we assume that $a \geqslant 0$ and sums up to 1.

Today's focus: quantifying distances between probability distributions

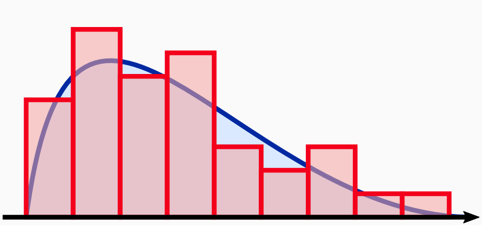


We must **handle** both **discrete** and **continuous** distributions.

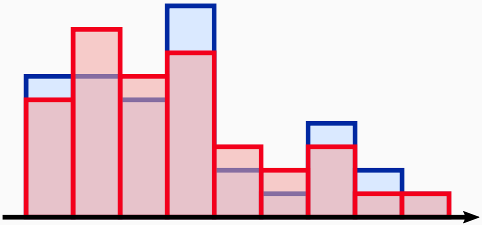


We must **choose** if α is closer to β (same mean value) or to γ (same support).

Application 1: One-sample and Two-sample testing



One-sample test:
discrete observation α,
continuous model β.

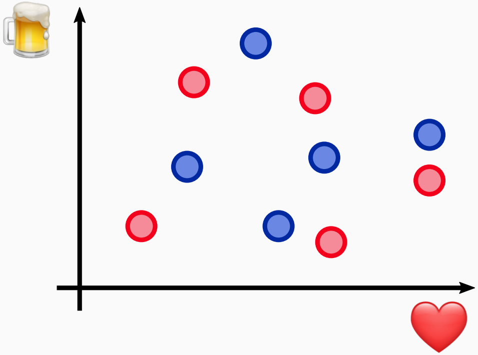


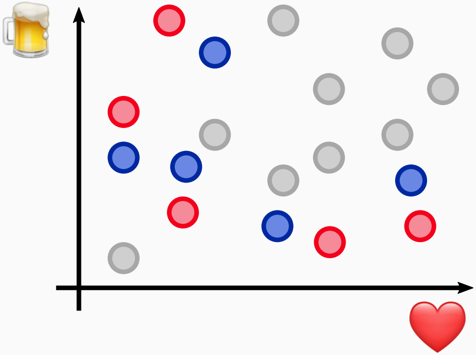
Two-sample test:
two discrete observations α and β.

Null hypothesis: α and β come from the **same distribution**.

Test: reject if $\mathrm{d}(\alpha, \beta)$ is too large.

Example: Splitting a population evenly for a clinical trial





Problem 1: ensure that the treatment and control groups have similar characteristics.

Problem 2: given a large population, pick a group of control patients that have similar characteristics to our treated patients.

Application 2: Classification = regression in a space of distributions

Linear regression:

- Encode class labels as **integer numbers**

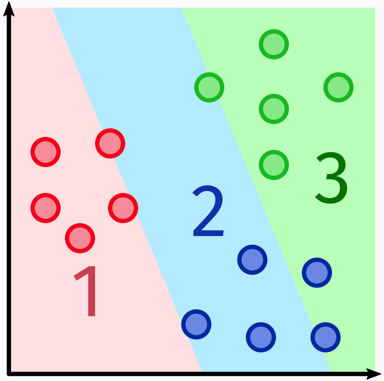
$$l(x) \in \{1, 2, 3\}\,.$$

- Predict a **score** $s(x)$ at every location x.

- Minimize the **least square error**:

$$\frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} \big| l(x) - s(x) \big|^2\,.$$

Massive **bias** depending on
the **ordering** of the labels.



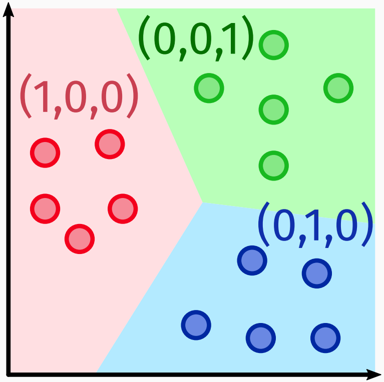
2 input features, 3 classes.

Logistic regression:

- Encode class labels as **probability distributions** $\delta(x) \in \mathbb{P}(\{1,2,3\})$.
- Predict a vector of **scores** $s_i(x)$ at every location x and turn it into a probability distribution using the **SoftMax**:
$\alpha(x) = (e^{s_1(x)}, e^{s_2(x)}, e^{s_3(x)}) \, / \sum e^{s_i(x)}$
- Minimize the **relative entropy**:

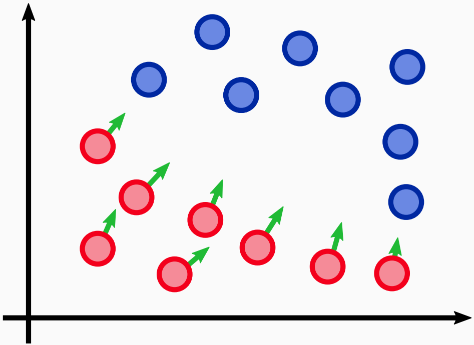
$$\frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} \mathrm{KL}(\delta(x), \alpha(x)) \, .$$

Invariant to the **ordering** of the labels.

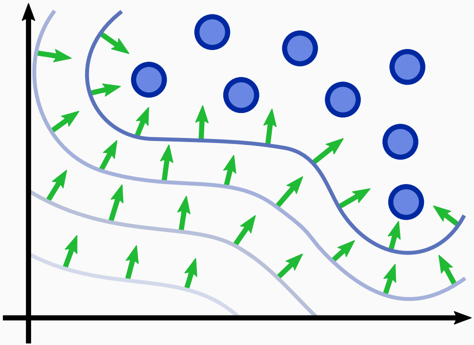


2 input features, 3 classes.

Application 3: Generative modelling

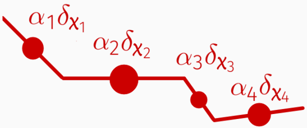


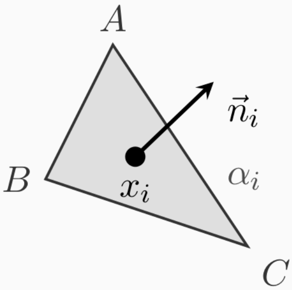
Generative Adversarial Networks and Variational Auto-Encoders **minimize a distance** between a synthetic sample and a reference data sample.

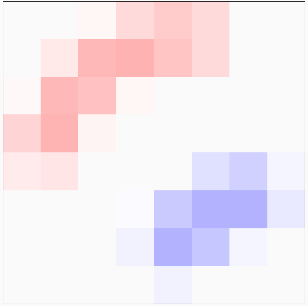


Diffusion and score-based models estimate a gradient of the **distance to the support** of a reference data sample.

Application 4: Shape registration [KCC17]







Curve:
one weight per **segment**.

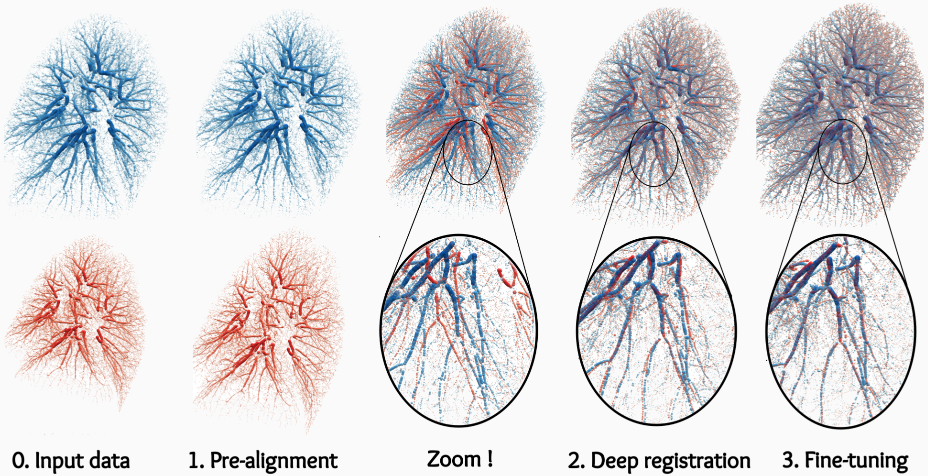
Surface:
one weight per **triangle**.

Segmentation mask:
one weight per **voxel**.

Encoding shapes as distributions guarantees an **invariance to resamplings**.

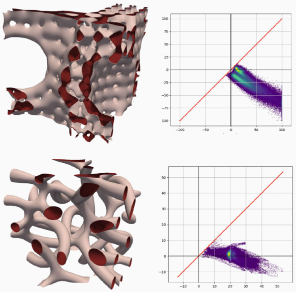
We may work with **basic** (x, y, z) coordinates or with **better features**.

Application 4: Shape registration [SFL+21]

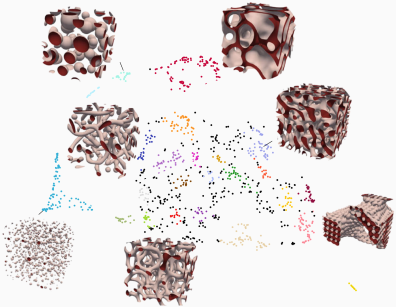


Registration algorithms **minimize a distance**
between a deformable model α and the fixed target β.

Application 5: Meta-analyses on histograms and distributions



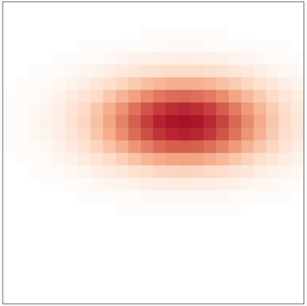
3D shape **texture**
$\simeq$ Distribution of **curvatures**
$\kappa_1 \geqslant \kappa_2$ on the surface.



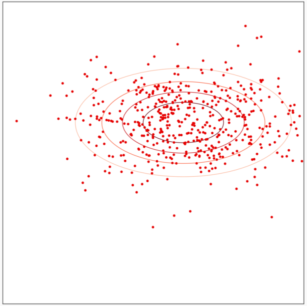
UMAP representation of a population of textures,
from the matrix of Wasserstein **distances between**
curvature histograms.

Distances enable the processing of **populations of histograms**.
This is relevant to make **group-level** analyses.

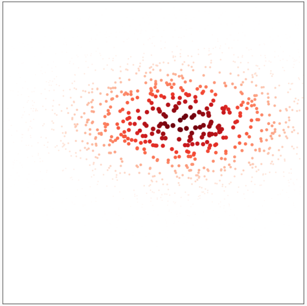
A point about implementations



Histogram:
explicit weights a_i,
implicit locations x_i.



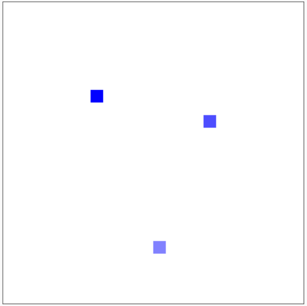
Sample:
implicit weights $1/N$,
explicit locations x_i.



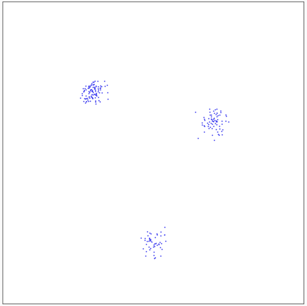
Weighted point cloud:
explicit weights a_i,
explicit locations x_i.

Depending on the application, we may choose a **different encoding** for our distributions.

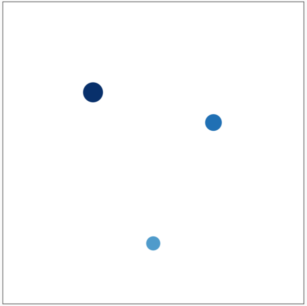
A point about implementations



Histogram:
explicit weights a_i,
implicit locations x_i.



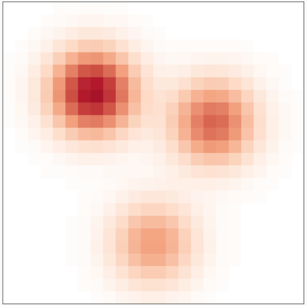
Sample:
implicit weights $1/N$,
explicit locations x_i.



Weighted point cloud:
explicit weights a_i,
explicit locations x_i.

Understanding that **different implementations** correspond to **the same operation** is key to insightful research in the field.

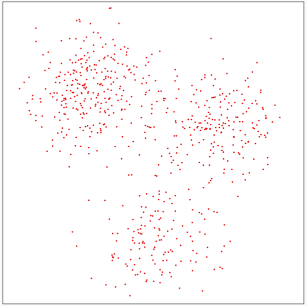
A point about implementations



Convolution

of the **density map** $a[\,i,\ j\,]$

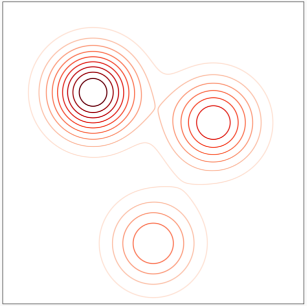
with a filter $g[\,i,\ j\,]$.



Additive noise:

$x_i \mapsto x_i + w_i$

where $w_i \sim \mathcal{N}(0, \sigma^2)$.



Soft distance:

log-likelihood $\ell(x) =$

$\log\big(\sum_i a_i e^{-\|x-x_i\|^2/2\sigma^2}\big)$.

Understanding that **different implementations** correspond to **the same operation** is key to insightful research in the field.

A point about notations

If $\alpha = \sum_{i=1}^{\mathsf{N}} a_i \delta_{x_i}$ is a **probability distribution**
and $f : x \mapsto f(x) \in \mathbb{R}$ is a **continuous function**,

$$\underbrace{\sum_{i=1}^{\mathsf{N}} a_i f(x_i)}_{\text{Programming}} \;=\; \underbrace{\int_x f(x)\,\mathrm{d}\alpha(x)}_{\text{Integration}} \;=\; \underbrace{\langle\, \alpha \,,\, f \,\rangle}_{\text{Analysis}} \;=\; \underbrace{\mathbb{E}_{X\sim\alpha}[f(X)]}_{\text{Probability}}\;.$$

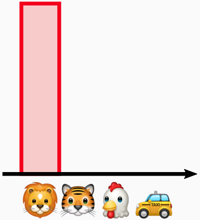
To study **spaces** of probability distributions,
the $\langle\, \alpha \,,\, f \,\rangle$ notation is **superior** as it highlights
the **linearity** with respect to **both** distributions and functions:

$$\begin{aligned} \langle\, \tfrac{1}{2}\alpha + \tfrac{1}{2}\beta \,,\, f \,\rangle \;&=\; \tfrac{1}{2}\langle\, \alpha \,,\, f \,\rangle + \tfrac{1}{2}\langle\, \beta \,,\, f \,\rangle\,, \\ \langle\, \alpha \,,\, f + g \,\rangle \;&=\; \langle\, \alpha \,,\, f \,\rangle + \langle\, \alpha \,,\, g \,\rangle\,. \end{aligned}$$

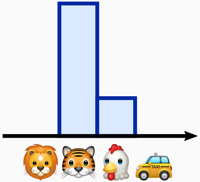
Major distances between distributions

Two main questions [Sav15]

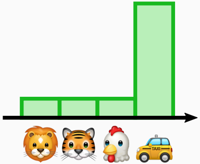
Image



Ground truth



Prediction 1



Prediction 2

When **designing** a distance between histograms:

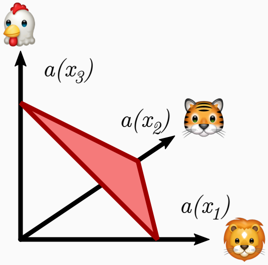
- Should we leverage the **distance** $\|x - y\|$ on the **“ground space” of labels**?
- How harshly should be we **penalize errors** on the estimation of the **support**?

The total variation distance

The space of probability distributions on

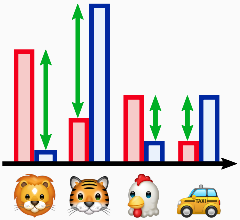
$$\{x_1, \dots, x_\mathsf{K}\}$$

is a **simplex** of dimension $\mathsf{K} - 1$.



The Total Variation is the L1–Manhattan distance:

$$\mathrm{TV}(\alpha, \beta) \;=\; \sum_i |a(x_i) - b(x_i)|\,.$$



This distance:

- **Maxes out** at 2 with **disjoint** supports.
- Pays no attention to $\|x_i - x_j\|$.
- Pays no attention to **unlikely events**.

Maximum likelihood and entropy

If $\beta = (b(1), \dots, b(\mathsf{K}))$ is a **model** distribution on $\{1, \dots, \mathsf{K}\}$, the likelihood of observing a sample x is $L_\beta(x) = b(x)$.

Assuming independence, the joint likelihood of a sample $(x_1, \dots, x_\mathsf{N})$ is:

$$L_\beta(x_1, \dots, x_\mathsf{N}) = b(x_1) \cdots b(x_\mathsf{N}) .$$

Finding a sample $(x_1, \dots, x_\mathsf{N})$ that **maximizes the likelihood** is equivalent to minimizing:

$$\ell_\beta(x_1, \dots, x_\mathsf{N}) = -\frac{1}{\mathsf{N}} \log \left[L_\beta(x_1, \dots, x_\mathsf{N}) \right] = \frac{1}{\mathsf{N}} \sum_{i=1}^{\mathsf{N}} \log \left[1/b(x_i) \right]$$

If the x_i are drawn **independently** according to a data distribution α, this converges to:

$$\ell_\beta(\alpha) = \lim_{\mathsf{N} \to +\infty} \sum_{k=1}^{\mathsf{K}} \frac{\#\{\, i \mid x_i = k \,\}}{\mathsf{N}} \log \left[1/b(k) \right] = \sum_{k=1}^{\mathsf{K}} a(k) \log \left[1/b(k) \right]$$

Maximum likelihood and entropy

In practice, the data distribution α is fixed and we try to find a model distribution β which is as likely as possible.

This is equivalent to minimizing the **relative entropy** or **Kullback–Leibler** divergence:

$$\mathrm{KL}(\alpha,\beta) \;=\; \ell_\beta(\alpha) - \ell_\alpha(\alpha) \;=\; \sum_{k=1}^{\mathsf{K}} a(k)\, \log\left[a(k)/b(k)\right].$$

We have that $\mathrm{KL}(\alpha,\alpha) = 0$ and $\mathrm{KL}(\alpha,\beta) \geqslant 0$, since $\log$ is concave:

$$\begin{aligned}
\log\left[b(k)/a(k)\right] \;&\leqslant\; b(k)/a(k) - 1 \\
\Longrightarrow \qquad \log\left[a(k)/b(k)\right] \;&\geqslant\; 1 - b(k)/a(k) \\
\Longrightarrow \sum_{k=1}^{\mathsf{K}} a(k)\, \log\left[a(k)/b(k)\right] \;&\geqslant\; \sum_{k=1}^{\mathsf{K}} a(k)\left[1 - b(k)/a(k)\right] \;=\; 0\,.
\end{aligned}$$

First properties of the relative entropy

$$\mathrm{KL}(\alpha, \beta) \;=\; \sum_{k=1}^{\mathsf{K}} a(k)\,\log\left[a(k)/b(k)\right] = \int_x a(x)\,\log\left[a(x)/b(x)\right]\mathrm{d}x:$$

- Is **not symmetric** – remember it as $\mathrm{KL}($data | model$)$.
- Is tied to an assumption of **independence**.
- Historically: compression on communication networks $\Longrightarrow$ .zip format.

Crucially, the relative entropy:

- Pays no attention to $\|x_i - x_j\|$.
- Pays **a lot** of attention to **unlikely events**: $\log(0^+) = -\infty$.

Information geometry: the Fisher–Rao metric on statistical manifolds [Fey17]

The Gauss map defines a **parametric surface**:

$$\mathcal{N} : (m, \sigma) \in \mathbb{R} \times \mathbb{R}_+ \mapsto \mathcal{N}(m, \sigma) \in \mathbb{P}(\mathbb{R}) \, .$$

Direct computations show that:

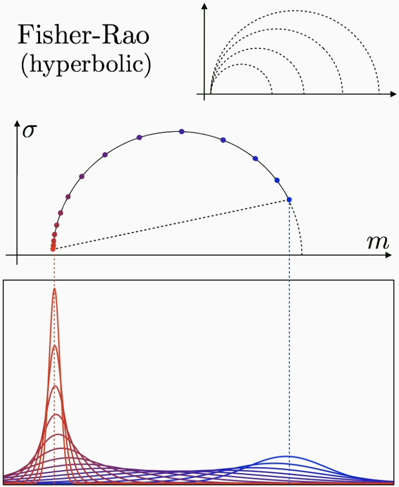
$$\mathrm{KL}(\mathcal{N}(m + \mathrm{d}m, \sigma + \mathrm{d}\sigma), \mathcal{N}(m, \sigma)) = \frac{\frac{1}{2}\mathrm{d}m^2 + \mathrm{d}\sigma^2}{\sigma^2} + o(\mathrm{d}m^2, \mathrm{d}\sigma^2)$$

$\Longrightarrow$ **Poincaré** metric on the upper half-plane.

With its **invariance to translation and scalings**,

the relative entropy induces a **hyperbolic** geometry

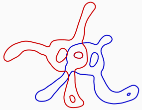
on the surface of Gaussian distributions.



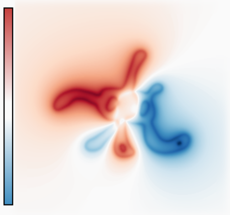
Kernel norms: recover compatibility with the addition

For sources $\alpha = \sum_i a_i \delta_{x_i}$ and targets $\beta = \sum_j b_j \delta_{y_j}$, choose a symmetric function g that induces a convolution kernel $k = g \star g$ and use:

$$\begin{aligned}
\mathrm{d}_k(\alpha, \beta) &= \| g \star \alpha - g \star \beta \|_{L^2}^2 \\
&= \langle \alpha - \beta \,,\, k \star (\alpha - \beta) \rangle \\
&= \sum_i \sum_j a_i a_j \, k(x_i, x_j) \\
&\quad - 2 \sum_i \sum_j a_i b_j \, k(x_i, y_j) \\
&\quad + \sum_i \sum_j b_i b_j \, k(y_i, y_j) \,.
\end{aligned}$$



Distributions α and β.



Blurred signal $g \star (\alpha - \beta)$.

Kernel norms: recover compatibility with the addition

Kernel norms (aka. Hilbert or Sobolev norms, Maximum Mean Discrepancies):

- Are **quadratic** with respect to the **weights** a_i and b_j.
- Are compatible with the addition – the geodesic from α to β is:

$$t \in [0,1] \mapsto (1-t)\,\alpha + t\,\beta\,.$$

- Have **wildly different behaviors** depending on $k(x,y)$: see the **lab session**.

Crucially, these formulas:

- Pay **a lot** of attention to $\|x_i - y_j\|$.
- Pay little attention to **unlikely events**,
  except if they are associated to **large values of** $k(x,y)$.

Optimal transport (OT) generalizes sorting to spaces of dimension $D > 1$

If $A = (x_1, \dots, x_N)$ and $B = (y_1, \dots, y_N)$
are two clouds of N points in $\mathbb{R}^D$, we define:

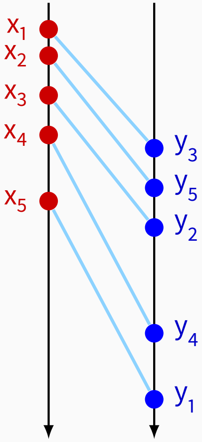
$$\mathrm{OT}(A, B) \;=\; \min_{\sigma \in \mathcal{S}_N} \frac{1}{2N} \sum_{i=1}^{N} \| x_i - y_{\sigma(i)} \|^2$$

Generalizes **sorting** to metric spaces.

Linear problem on the permutation matrix P:

$$\mathrm{OT}(\alpha, \beta) \;=\; \min_{P \in \mathbb{R}^{N \times N}} \sum_{i,j=1}^{N} P_{i,j} \cdot \tfrac{1}{2} \| x_i - y_j \|^2 ,$$

$$\text{s.t.} \quad P_{i,j} \geqslant 0 \quad \underbrace{\textstyle\sum_j P_{i,j} = a_i}_{\text{Each source point...}} \quad \underbrace{\textstyle\sum_i P_{i,j} = b_j}_{\text{is transported onto the target.}} .$$



assignment

$\sigma : [\![1, 5]\!] \to [\![1, 5]\!]$

Practical use

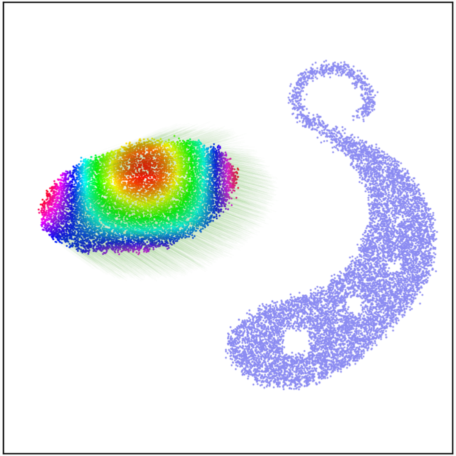
Alternatively, we understand OT as:

- Nearest neighbor **projection** + **incompressibility** constraint.
- Fundamental example of **linear optimization** over the transport plan $\mathsf{P}_{i,j}$.

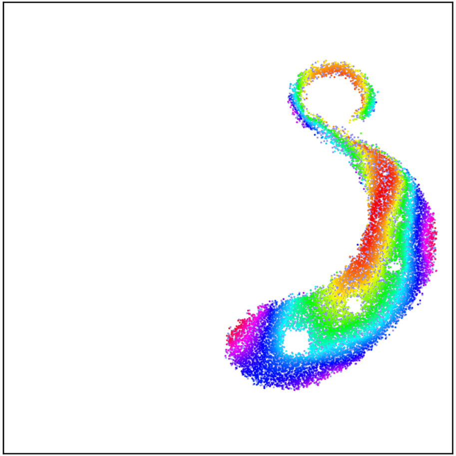
This theory induces two main quantities:

- The transport plan $\mathsf{P}_{i,j} \simeq$ the optimal mapping $x_i \mapsto y_{\sigma(i)}$.
- The "Wasserstein" distance $\sqrt{\mathrm{OT}(\mathsf{A}, \mathsf{B})}$.

The optimal transport plan

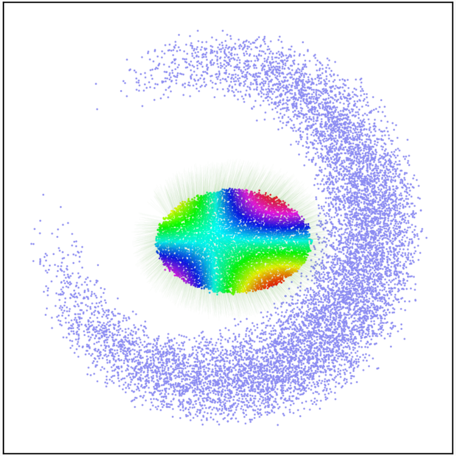


Before

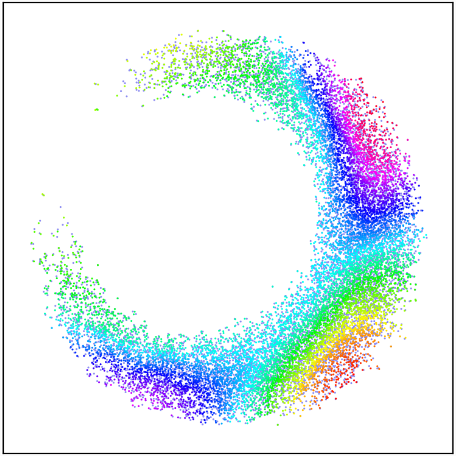


After

The optimal transport plan

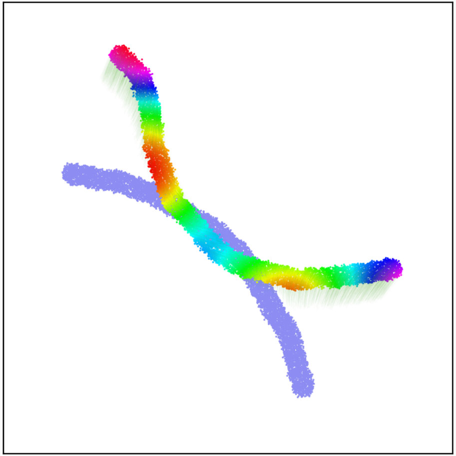


Before



After

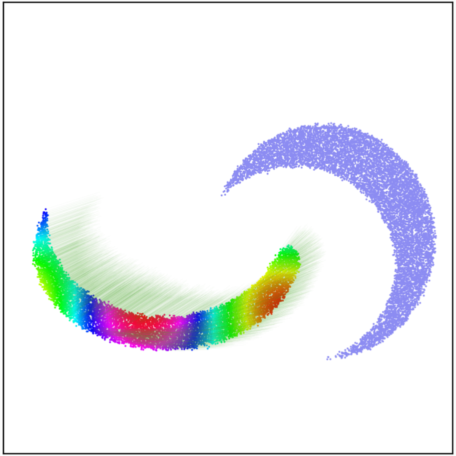
The optimal transport plan



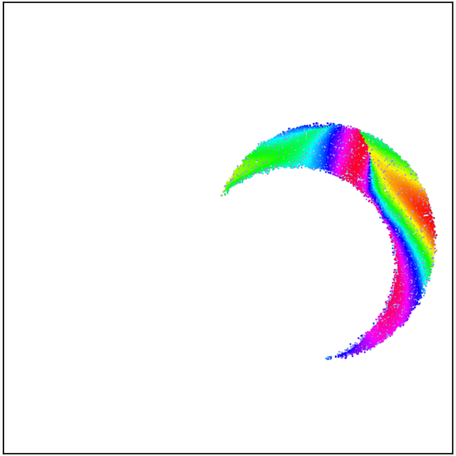
Before

After

The optimal transport plan



Before



After

The Wasserstein metric on statistical manifolds [PC18]

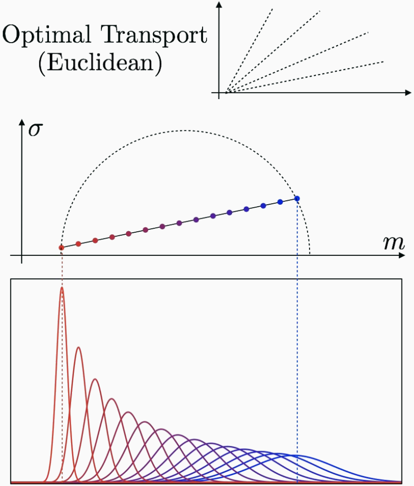
The Gauss map defines a **parametric surface**:

$$\mathcal{N} : (m, \sigma) \in \mathbb{R} \times \mathbb{R}_+ \mapsto \mathcal{N}(m, \sigma) \in \mathbb{P}(\mathbb{R}) \, .$$

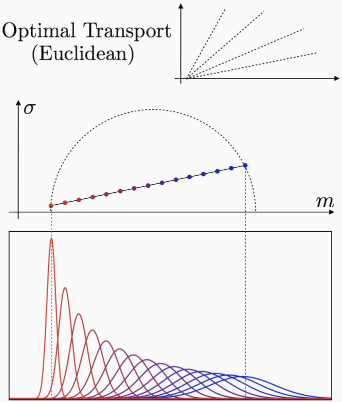
Direct computations show that:

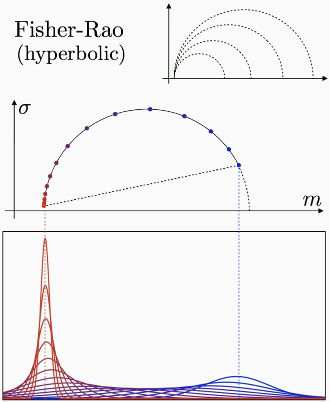
$$\begin{aligned} &2\,\mathrm{OT}(\mathcal{N}(m_1, \sigma_1), \mathcal{N}(m_2, \sigma_2)) \\ &= (m_1 - m_2)^2 \; + \; (\sigma_1 - \sigma_2)^2 \, . \end{aligned}$$

$\Longrightarrow$ **Euclidean** metric on the upper half-plane.
Optimal transport **lifts the geometry of the sample space** to the surface of Gaussian distributions.



Two canonical distances between Gaussian distributions [PC18]





Gaussians + **Wasserstein** metric

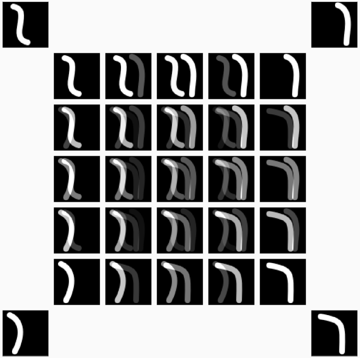
$=$ **Euclidean**.

Gaussians + relative **entropy**

$=$ **Poincaré**.

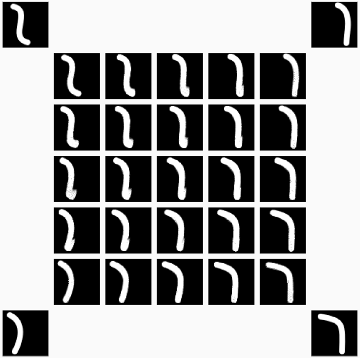
Geometric solutions to least square problems [AC11]

$$\text{Barycenter } \mathsf{A}^* = \arg\min_{\mathsf{A}} \sum_{i=1}^{4} \lambda_i \, \text{Loss}(\mathsf{A}, \mathsf{B}_i) .$$



Euclidean barycenters.

$$\text{Loss}(\mathsf{A}, \mathsf{B}) = \|\mathsf{A} - \mathsf{B}\|_{L^2}^2$$



Wasserstein barycenters.

$$\text{Loss}(\mathsf{A}, \mathsf{B}) = \mathrm{OT}(\mathsf{A}, \mathsf{B})$$

How should we solve the OT problem?

Key dates for discrete optimal transport with N points:

- [Kan42]: **Dual** problem of Kantorovitch.
- [Kuh55]: **Hungarian** methods in $O(\mathrm{N}^3)$.
- [Ber79]: **Auction** algorithm in $O(\mathrm{N}^2)$.
- [KY94]: **SoftAssign** = Sinkhorn + simulated annealing, in $O(\mathrm{N}^2)$.
- [GRL$^+$98, CR00]: **Robust Point Matching** = Sinkhorn as a loss.
- [Cut13]: Start of the **GPU era.**
- [Mér11, Lév15, Sch19]: **multi-scale** solvers in $O(\mathrm{N}\log\mathrm{N})$.

- **Solution,** today: **Multiscale Sinkhorn algorithm, on the GPU**.

$\Longrightarrow$ Generalized **QuickSort** algorithm,

$\simeq O(\mathrm{N}\log\mathrm{N})$ if D is small, fast $O(\mathrm{N}^2)$ otherwise.

Scaling up optimal transport to anatomical data

Progresses of the last decade add up to a $\times$**100** - $\times$**1000** acceleration:

$$\text{Sinkhorn GPU} \xrightarrow{\times 10} + \text{KeOps} \xrightarrow{\times 10} + \text{Annealing} \xrightarrow{\times 10} + \text{Multi-scale}$$

With a precision of 1%, on a gaming GPU:

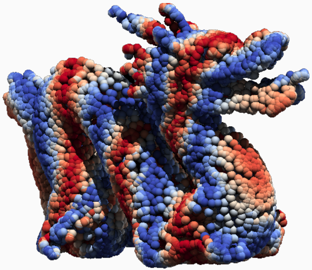
`pip install geomloss`

+

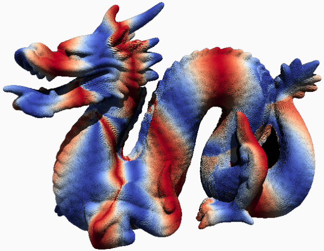
modern GPU

(1 000 €)

$\Longrightarrow$



10k points in 30-50ms



100k points in 100-200ms

Recap on classical distances between probability distributions

The **Total Variation**:

- **Invariant** to the ground metric $\|x_i - y_j\|$, only cares about **large** weights a_i and b_j.

The **relative entropy** KL:

- **Invariant** to the ground metric $\|x_i - y_j\|$, cares about the **ratio** $a_i \,/\, b_j$.

Kernel norms:

- More or less **faithful** to the ground metric **depending** on k, easy to scale on GPUs.

Optimal transport distances:

- **Extremely faithful** to the ground metric $\|x_i - y_j\|$.
- **Scalability** is recent – still open on general graphs and manifolds.

Open problem 1: Topology-aware distances

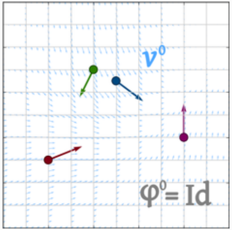
“OT that preserves the neighborhood structure”?

The problem has been studied for decades:

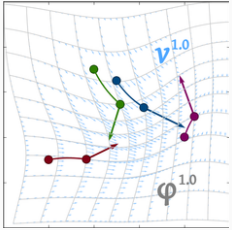
- **Optimization:** Quadratic assignment…
- **Optimal Transport:** Gromov–Wasserstein…
- **Fluid mechanics:** Camassa–Holm equation…
- **Shape analysis:** LDDMM, SVF…
- **Statistics:** Stein Variational Gradient Descent…
- **Deep learning:** Neural ODEs…

Mature tools exist but remain

$\geqslant$ 100 **slower** than Optimal Transport.



Initial configuration.



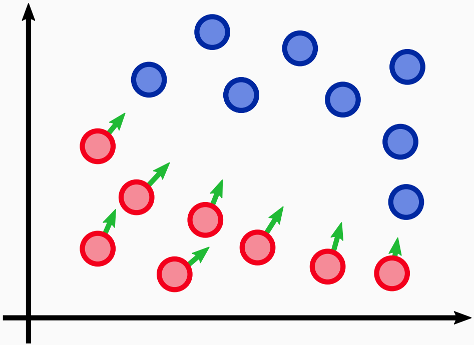
Diffeomorphic displacement.

Open problem 2: The curse of dimensionality

In **high dimension**, the matrix of Euclidean distances stops being informative.

Standard kernels and OT metrics are overwhelmed by **statistical noise**.

How can we compute **meaningful distances and gradients**?



GANs and VAEs **minimize a distance** between a synthetic sample and a reference data sample.

Dual norms: a fundamental insight from functional analysis

$$\text{Loss}(\alpha, \beta) \;=\; \max_{f \in B} \; \langle\, \alpha - \beta\,,\; f\,\rangle,$$

$$\text{look for } \theta^* \;=\; \arg\min_{\theta} \max_{f \in B} \; \langle\, \alpha(\theta) - \beta\,,\; f\,\rangle$$

- $B = \{\, \|\, f\,\|_\infty \leqslant 1\,\} \implies$ Loss $=$ TV norm:
    - Zero geometry, always saturates on disjoint samples.
    - **Too many** test functions.
- $B = \{\, \|\, f\,\|^2_{L^2} + \|\, \nabla f\,\|^2_{L^2} + \cdots \leqslant 1\,\} \implies$ Loss $=$ kernel norm:
    - Screening artifacts – see lab session.
    - In high dimension, samples are at equal distance from each other.
      "Smooth" functions are either "constant" or **"bounded"**: fall back on TV behavior.

Dual norms: link with the GANs literature

$$\text{Loss}(\alpha, \beta) \;=\; \max_{f \in B} \; \langle\, \alpha - \beta \,,\; f \,\rangle,$$

$$\text{look for } \theta^* \;=\; \arg\min_{\theta} \max_{f \in B} \; \langle\, \alpha(\theta) - \beta \,,\; f \,\rangle$$

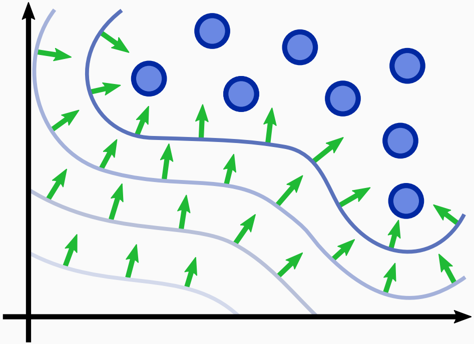
- $B = \{\, f$ is 1-Lipschitz $\} \implies$ Loss $=$ Wasserstein-1:
    - Modern solvers are nearly as efficient as a **closed formula**.
    - **Useless** in $(\mathbb{R}^{512\times512}, \|\cdot\|_2)$: the ground cost makes no sense.
- $B \simeq \{\, f$ is 1-Lipschitz $\} \bigcap \{\, f$ is a **CNN** $\} \implies$ Loss $=$ Wasserstein–GAN :
    - Use **perceptual** test functions.
    - No simple formula: use **gradient ascent**.
      Leads to a cumbersome min-max optimization.

Open problem 2: Understand the impact of domain-specific test functions f

Similar story for **diffusion models**: we use **CNNs** (U-Nets…) to predict the gradient of the distance to the set of **natural images**.

The **interplay** between **mathematical insights** derived from toy models and **numerical experiments** on modern hardware is at the **heart of ML research**.

Let's play with gradient descent to **build an intuition** about classical formulas!



Diffusion and score-based models estimate a gradient of the **distance to the support** of a reference data sample.

References

References i

M. Agueh and G. Carlier.

Barycenters in the Wasserstein space.

SIAM Journal on Mathematical Analysis, 43(2):904–924, 2011.

Dimitri P Bertsekas.

A distributed algorithm for the assignment problem.

Lab. for Information and Decision Systems Working Paper, M.I.T., Cambridge, MA, 1979.

References ii

Haili Chui and Anand Rangarajan.

A new algorithm for non-rigid point matching.

In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 2, pages 44–51. IEEE, 2000.

Marco Cuturi.

Sinkhorn distances: Lightspeed computation of optimal transport.

In *Advances in Neural Information Processing Systems*, pages 2292–2300, 2013.

References iii

Jean Feydy.

Data science workshop notes.

https://www.jeanfeydy.com/Teaching/data_science_workshop_notes.pdf, 2017.

Session 12.

Jean Feydy.

***Geometric data analysis, beyond convolutions*.**

PhD thesis, Université Paris-Saclay, 2020.

References iv

Steven Gold, Anand Rangarajan, Chien-Ping Lu, Suguna Pappu, and Eric Mjolsness.

New algorithms for 2d and 3d point matching: Pose estimation and correspondence.

Pattern recognition, 31(8):1019–1031, 1998.

Leonid V Kantorovich.

On the translocation of masses.

In *Dokl. Akad. Nauk. USSR (NS)*, volume 37, pages 199–201, 1942.

References v

Irene Kaltenmark, Benjamin Charlier, and Nicolas Charon.

A general framework for curve and surface comparison and registration with oriented varifolds.

In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3346–3355, 2017.

Harold W Kuhn.

The Hungarian method for the assignment problem.

Naval research logistics quarterly, 2(1-2):83–97, 1955.

Jeffrey J Kosowsky and Alan L Yuille.

The invisible hand algorithm: Solving the assignment problem with statistical physics.

Neural networks, 7(3):477–490, 1994.

Bruno Lévy.

A numerical algorithm for l2 semi-discrete optimal transport in 3d.

ESAIM: Mathematical Modelling and Numerical Analysis, 49(6):1693–1715, 2015.

References vii

Quentin Mérigot.

A multiscale approach to optimal transport.

In *Computer Graphics Forum*, volume 30, pages 1583–1592. Wiley Online Library, 2011.

Gabriel Peyré and Marco Cuturi.

Computational optimal transport.

arXiv preprint arXiv:1803.00567, 2018.

A. Savin.

Lion at the berlin zoo.

https://commons.wikimedia.org/wiki/File:Berlin_Tierpark_Friedrichsfelde_12-2015_img18_Indian_lion.jpg, 2015.

Art Libre.

Bernhard Schmitzer.

Stabilized sparse scaling algorithms for entropy regularized transport problems.

SIAM Journal on Scientific Computing, 41(3):A1443–A1481, 2019.

Zhengyang Shen, Jean Feydy, Peirong Liu, Ariel H Curiale, Ruben San Jose Estepar, Raul San Jose Estepar, and Marc Niethammer.

Accurate point cloud registration with robust optimal transport.

Advances in Neural Information Processing Systems, 34:5373–5389, 2021.