Aligning embeddings and geometric random graphs

The Procrustes-Wasserstein problem

Aligning embeddings and geometric random graphs

The Procrustes-Wasserstein problem

Mathieu Even,
Inria Montpellier

Luca Ganassali,
Paris Saclay

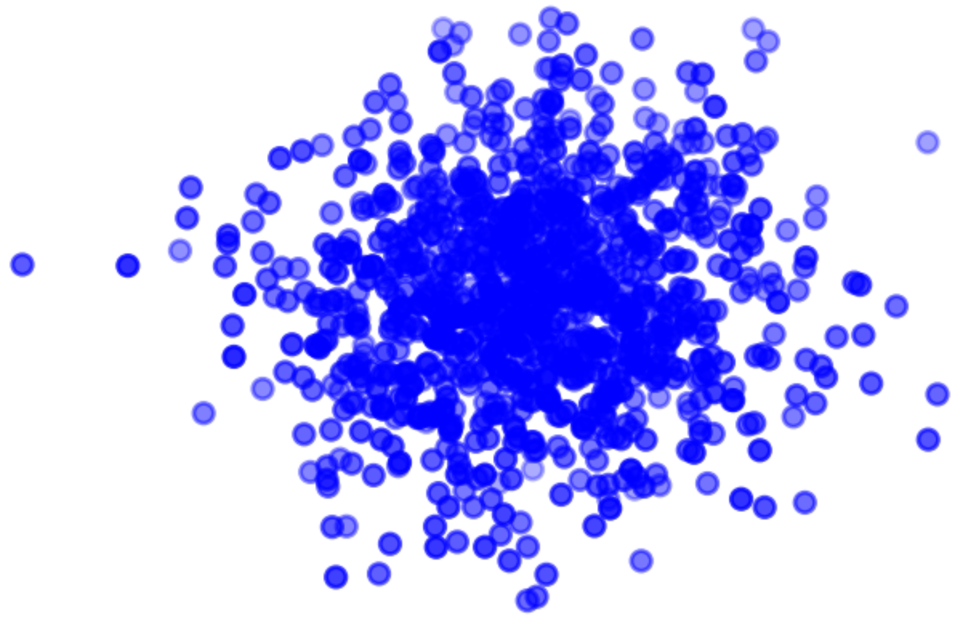
Jakob Maier,
Inria Paris

Laurent Massoulié,
Inria Paris

Aligning embeddings

Which x_i corresponds to which y_j?

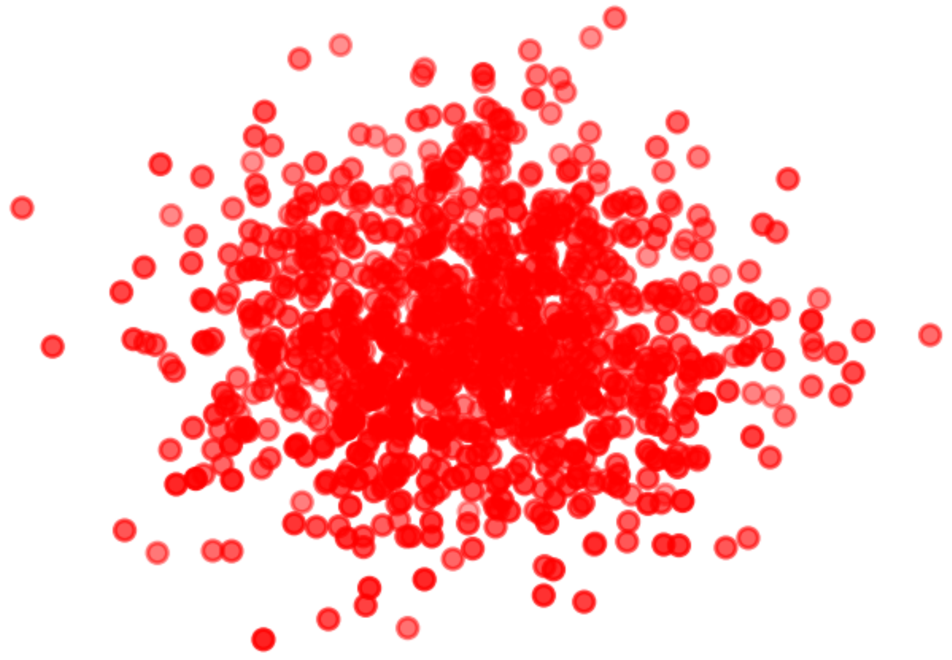
$\Longrightarrow$ find a permutation $\pi \in \mathcal{S}_n$ with $\pi(i) = j$



$\mathcal{X} = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$

Applications:

- Embeddings of different LLMs
- 3D object analysis



$\mathcal{Y} = \{y_1, \ldots, y_n\} \subset \mathbb{R}^d$

If the embeddings are rotation-invariant?

$\Longrightarrow$ find an orthogonal transformation

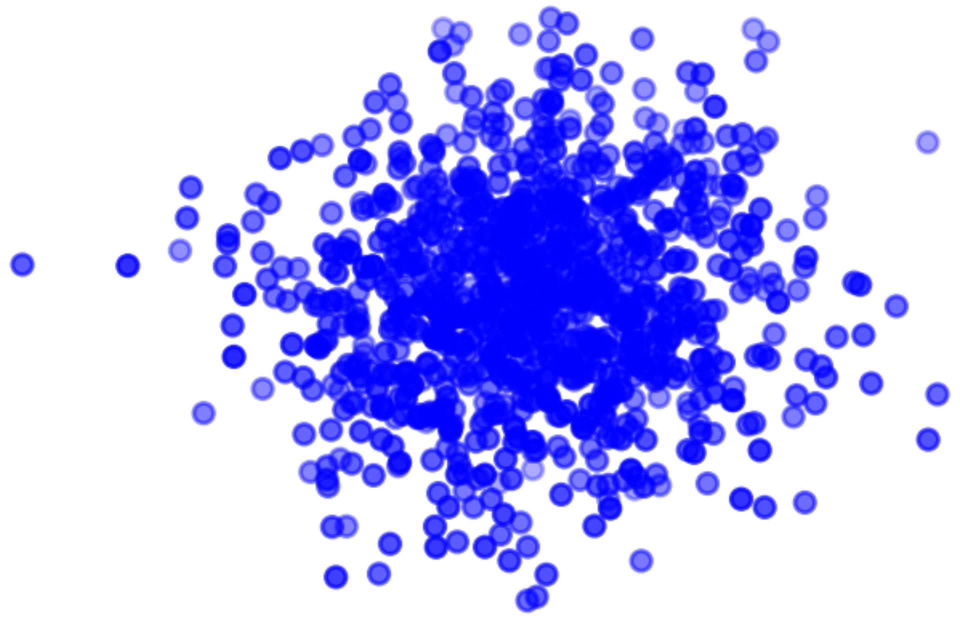
$Q \in \mathcal{O}(d)$ with $Qx_i = y_j$

Aligning embeddings

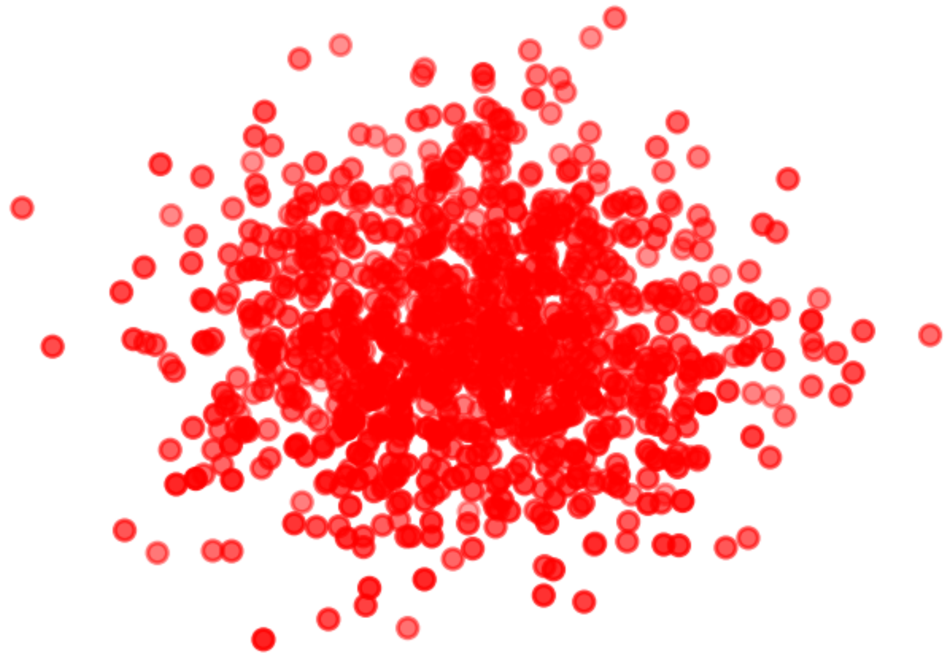
Goal:

Simultaneously find optimal π and Q:

$$\min_{\substack{\pi \in \mathcal{S}_n \\ Q \in \mathcal{O}_d}} \sum_{i=1}^{n} \|Qx_i - y_{\pi(i)}\|$$



$\mathcal{X} = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$



$\mathcal{Y} = \{y_1, \ldots, y_n\} \subset \mathbb{R}^d$

Algorithm idea 1:

Find initial π, then Ping-Pong

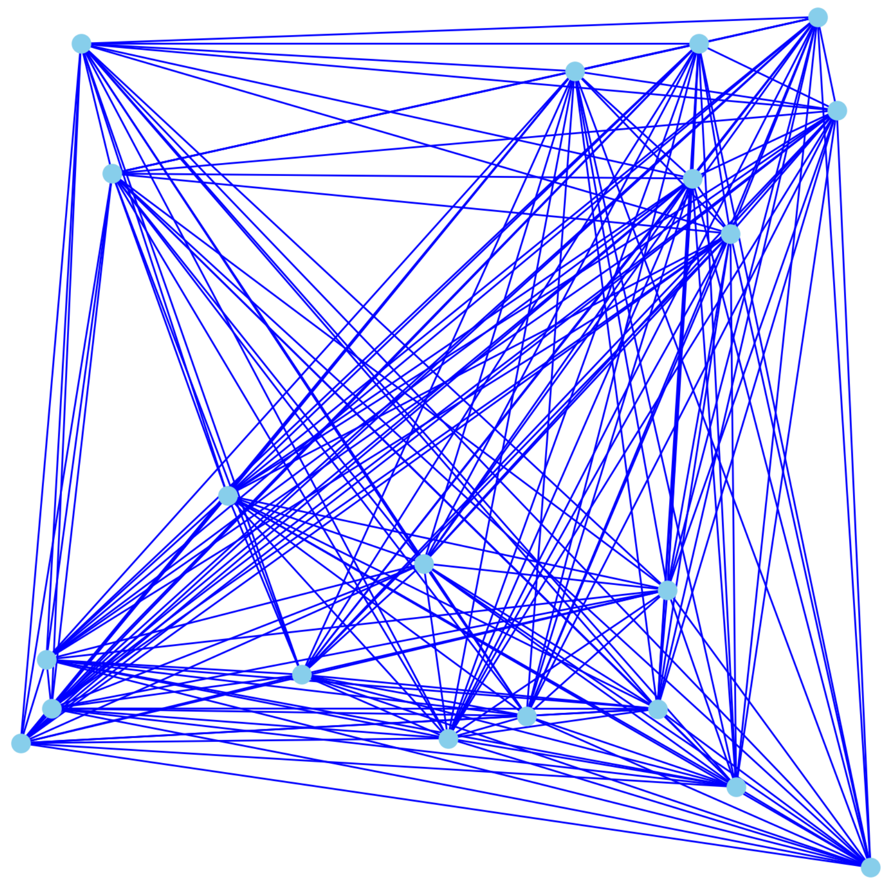
Easy if π is known:

Finding Q requires a single SVD.

Easy if Q is known:

Finding π is the LAP, solved in $\mathcal{O}(n^3)$

Aligning random geometric graphs



Graph A, weight matrix

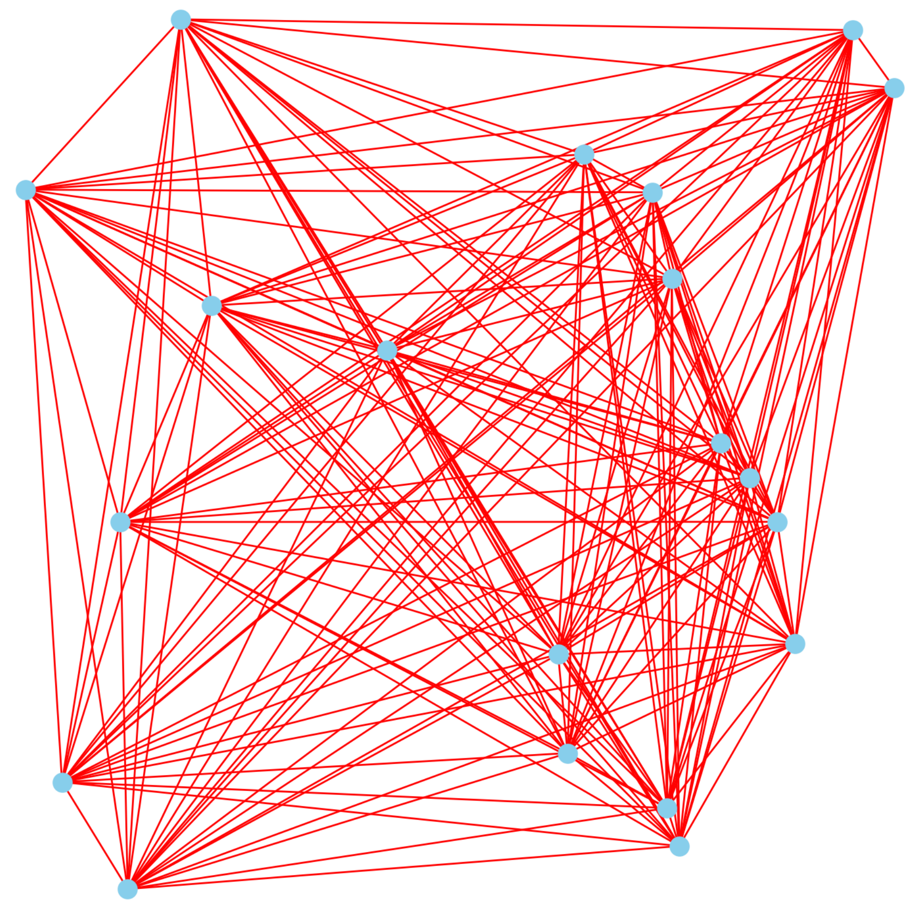
$A_{ij} = \langle x_i, x_j \rangle$

invariant under
orthogonal transformation

Goal:
Find the optimal node permutation to conserve edge weights:

$$\min_{\pi \in \mathcal{S}_n} \sum_{i,j=1}^{n} |A_{ij} - B_{\pi(i)\pi(j)}|^2$$

Algorithm idea 2:
Initialize Ping-Pong with doubly stochastic approximation of π



Graph B, weight matrix:

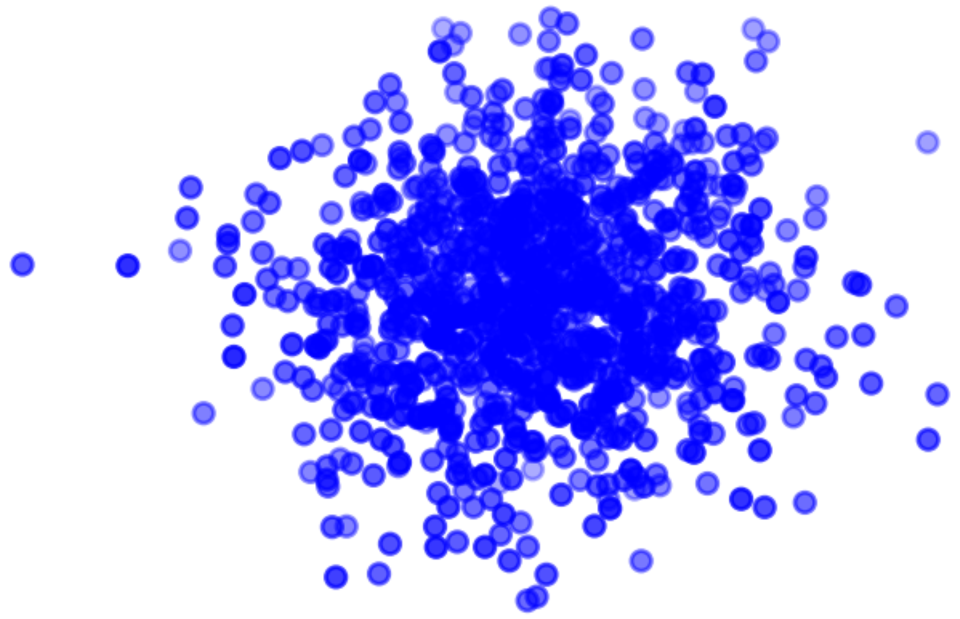
$B_{ij} = \langle y_i, y_j \rangle$

Stochastic setting

Goal: retrieve $\pi^\star$ and $Q^\star$

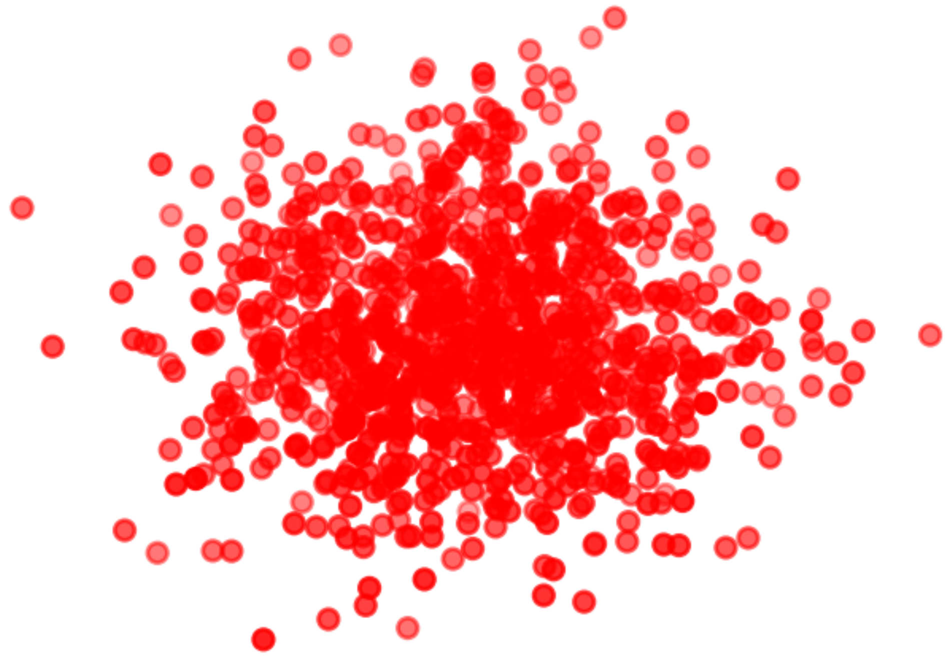
The MLE estimator of these quantities is

$$\hat{\pi}, \hat{Q} \in \arg \min_{\substack{\pi \in \mathcal{S}_n \\ Q \in \mathcal{O}_d}} \sum_{i=1}^{n} \|Qx_i - y_{\pi(i)}\|$$



$$\mathcal{X} = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$$

$$x_i \sim \mathcal{N}(0, I_d)$$



$$\mathcal{Y} = \{y_1, \ldots, y_n\} \subset \mathbb{R}^d$$

$$y_{\pi^\star(i)} = Q^\star x_i + \mathcal{N}(0, \sigma^2)$$

Performance metrics:

$$\text{overlap}(\hat{\pi}, \pi^\star) = \frac{1}{n} \left| \{ i : \hat{\pi}(i) = \pi^\star(i) \} \right|$$

$$c^2(\hat{\pi}, \pi^\star) = \frac{1}{n} \sum_{i=1}^{n} \|x_{\hat{\pi}(i)} - x_{\pi^\star(i)}\|^2$$

L^2 **transport cost**

Informational results

$$x_i \sim \mathcal{N}(0, I_d)$$

$$y_{\pi^\star(i)} = Q^\star x_i + \mathcal{N}(0, \sigma^2)$$

Low dimension $d \ll \log(n)$

Result from (Wang et al., 2024):

$\mathrm{overlap}(\hat{\pi}, \pi^\star) \to 1$ requires $\sigma \ll \frac{1}{n^{\frac{1}{d}}}$.

Our result:

$c^2(\hat{\pi}, \pi^\star) \in o(d)$ requires only $\sigma \ll \frac{1}{\sqrt{d}}$.

High dimension $d \gg \log(n)$

Our result: If $\sigma \to 0$, then

$\mathrm{overlap}(\hat{\pi}, \pi^\star) \to 1$

and

$c^2(\hat{\pi}, \pi^\star) \in o(d)$.

Aligning embeddings and geometric random graphs

Take a peak!