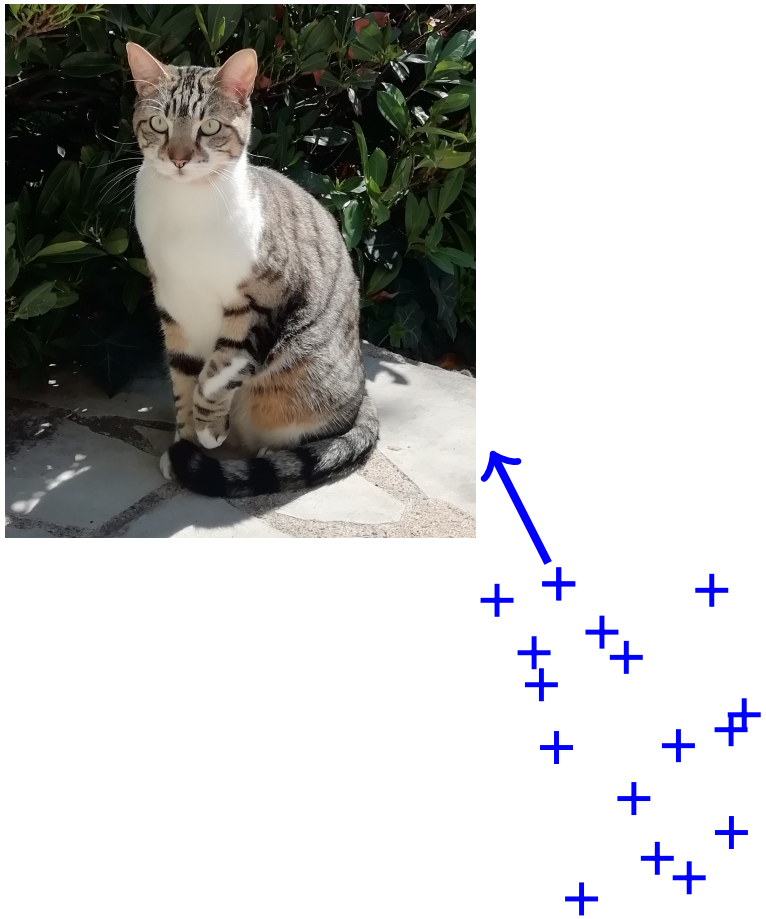
Contributions to Optimal Transport for Machine Learning: Ground Metric and Generalized Framework

Thesis defense of Tanguy KERDONCUFF

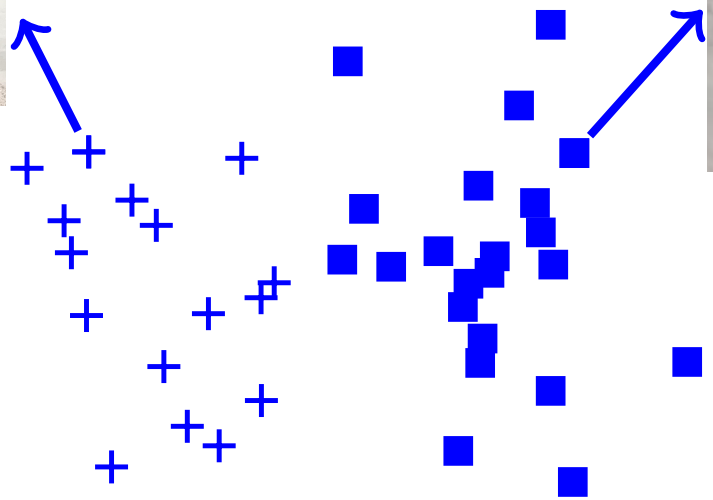
Élisa FROMONT	Professeure, Université de Rennes 1	Présidente
Marianne CLAUSEL	Professeure, Université de Lorraine	Rapporteuse
Nicolas COURTY	Professeur, Université Bretagne Sud	Rapporteur
Marc SEBBAN	Professeur, Université de Saint-Étienne	Directeur
Rémi EMONET	Maître de conférences, Université de Saint-Étienne	Co-encadrant

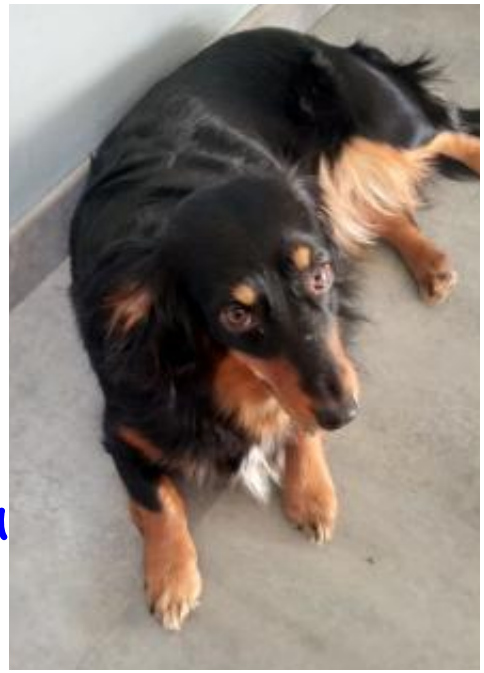
December 9, 2021

Machine Learning: a classification example

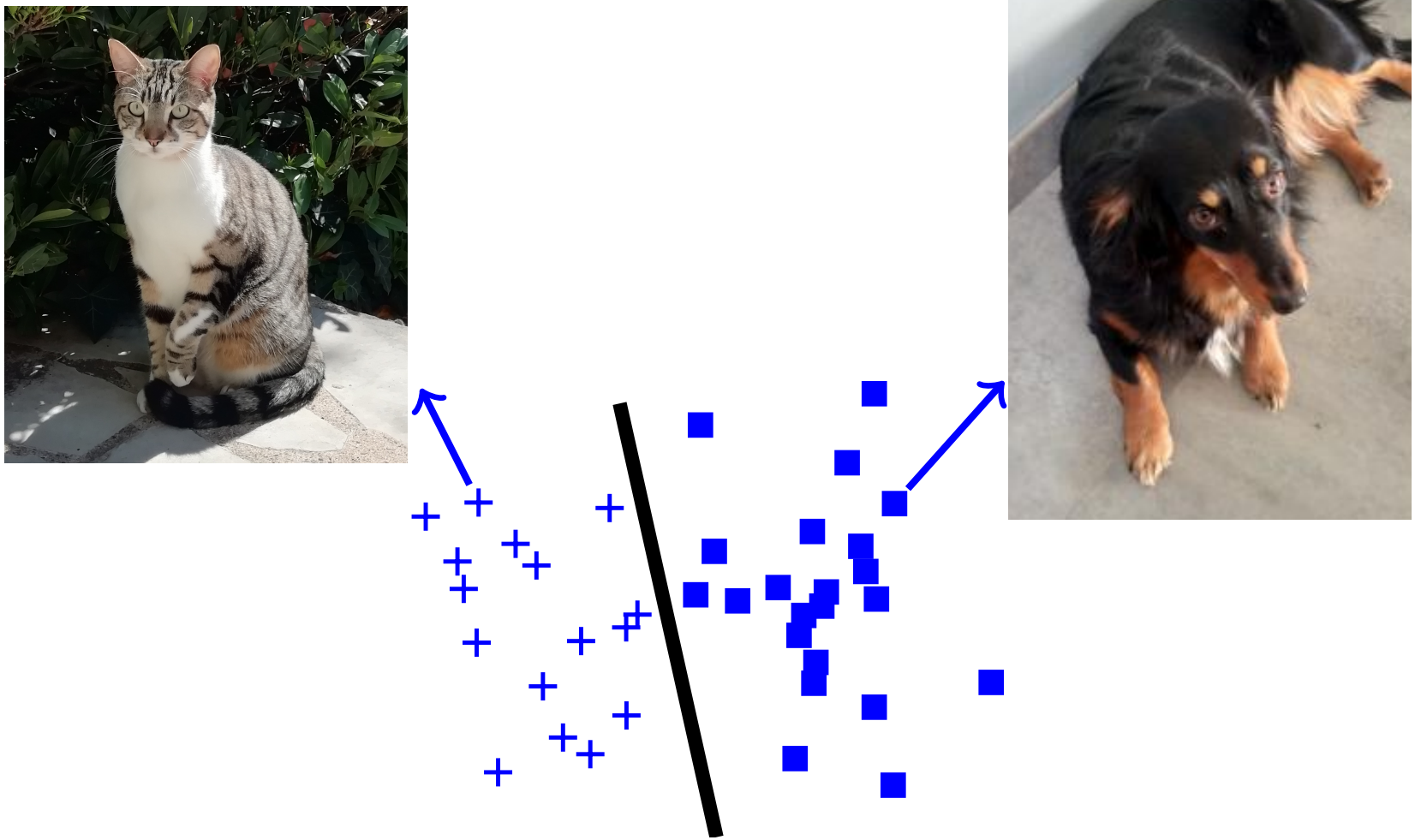


Machine Learning: a classification example

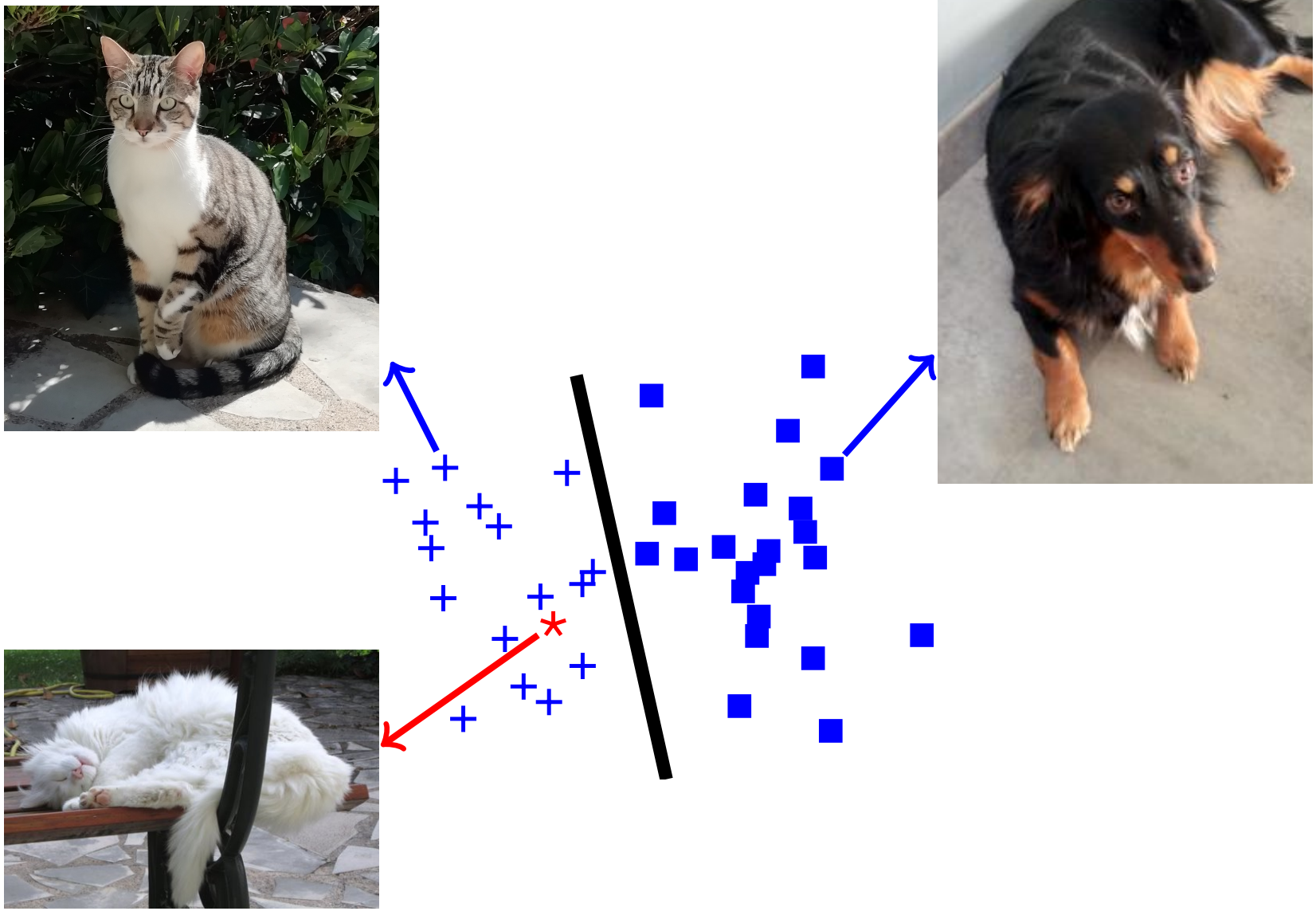




Machine Learning: a classification example



Machine Learning: a classification example



Intuition behind Optimal Transport

Intuition behind Optimal Transport

Intuition behind Optimal Transport

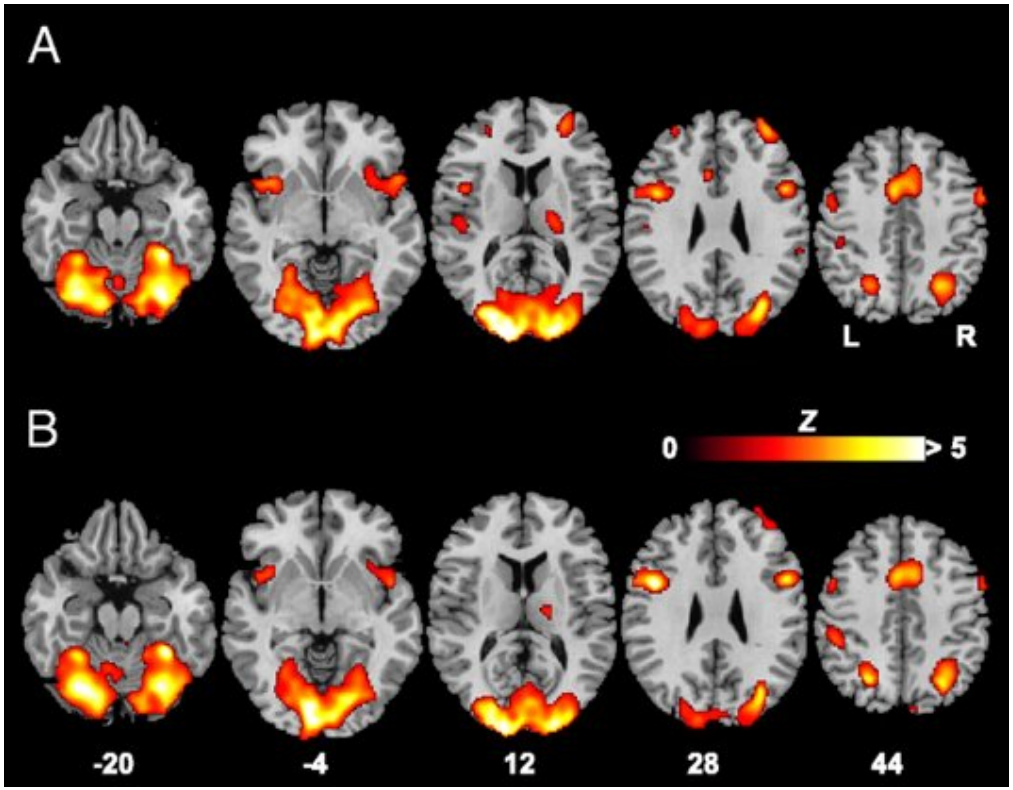
Intuition behind Optimal Transport

Intuition behind Optimal Transport

Intuition behind Optimal Transport

Optimal Transport in Machine Learning

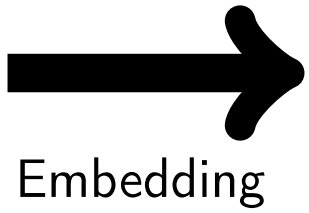
- We can use the Optimal Transport to compare two brain activation maps [Tan et al., 2008]

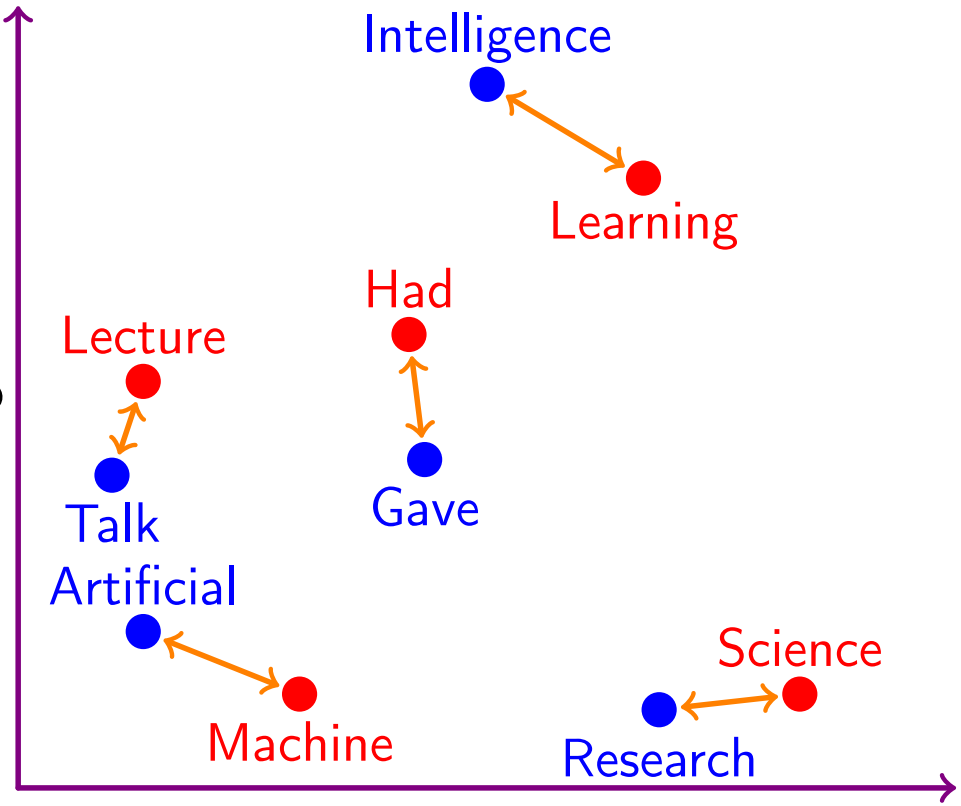


Optimal Transport in Machine Learning

- We can use the Optimal Transport to compare two sentences

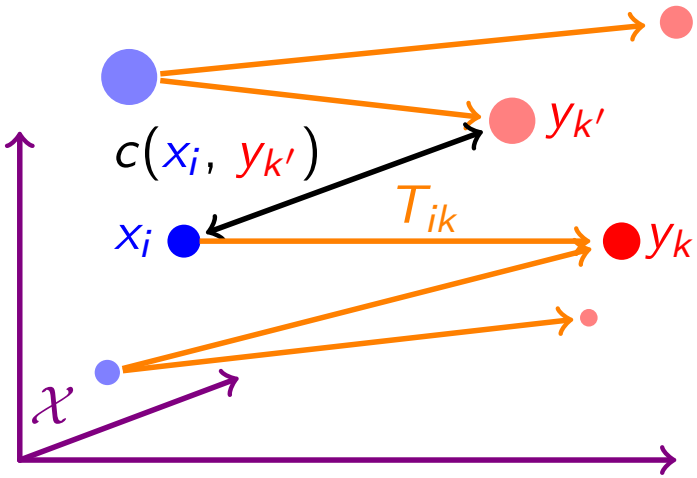
he	she
Gave	Had
a	a
Research	Science
Talk	Lecture
on	in
Artificial	Machine
Intelligence	Learning





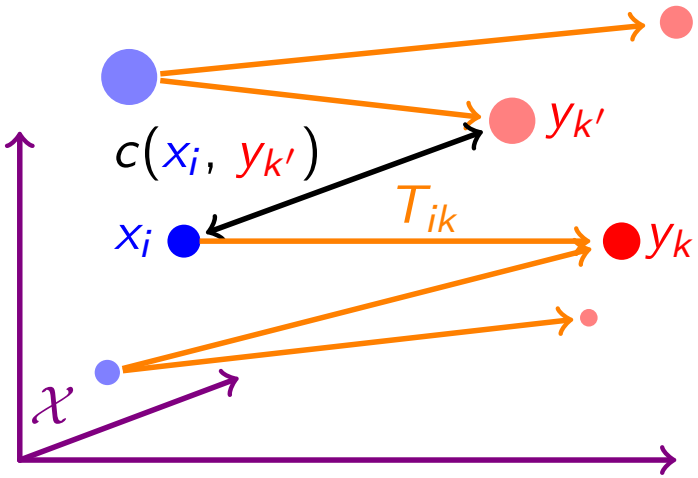
Overview

- Ground Metric

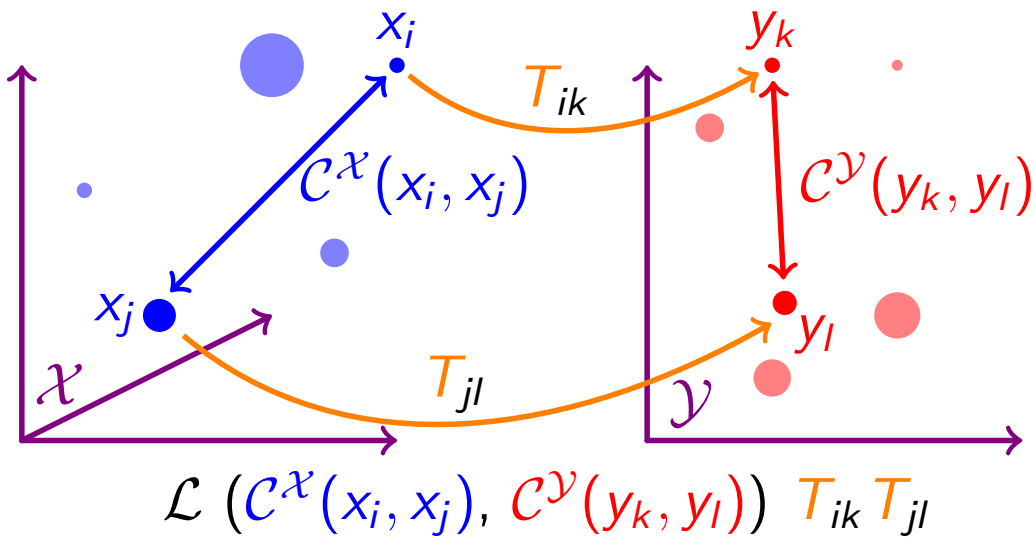


Overview

- Ground Metric

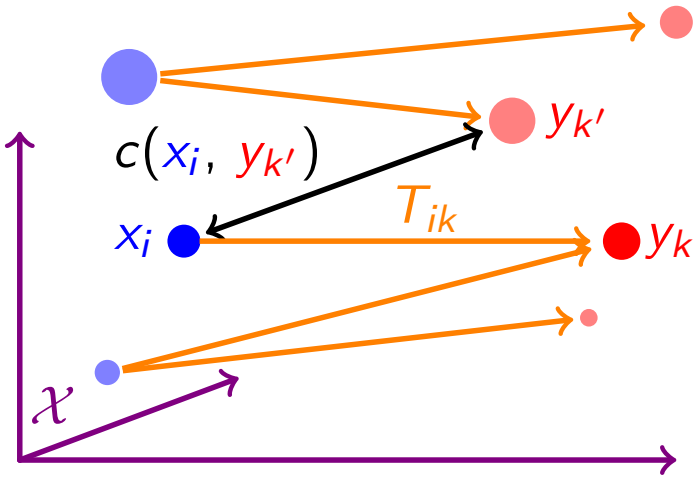


- Gromov Wasserstein

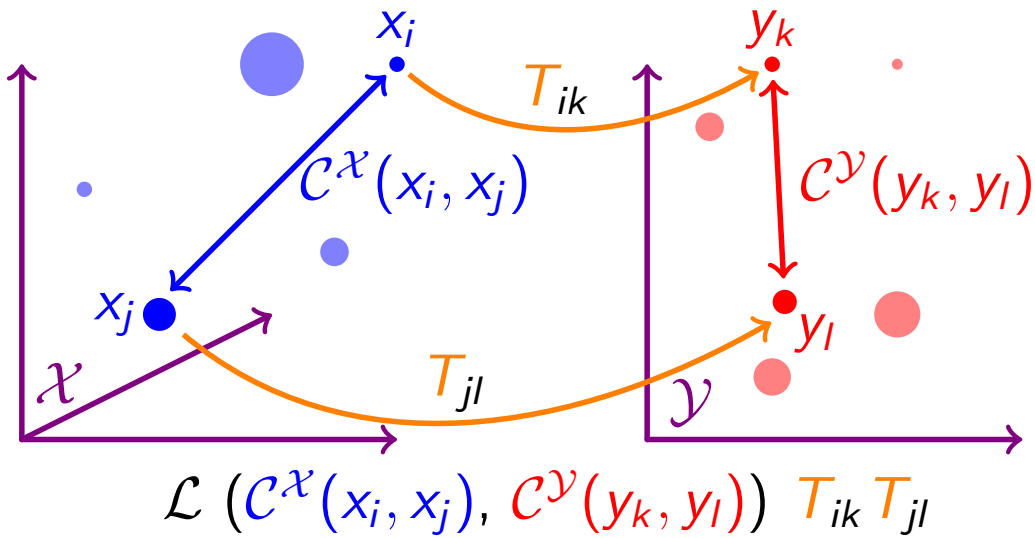


Overview

- Ground Metric



- Gromov Wasserstein



- Generalized OT

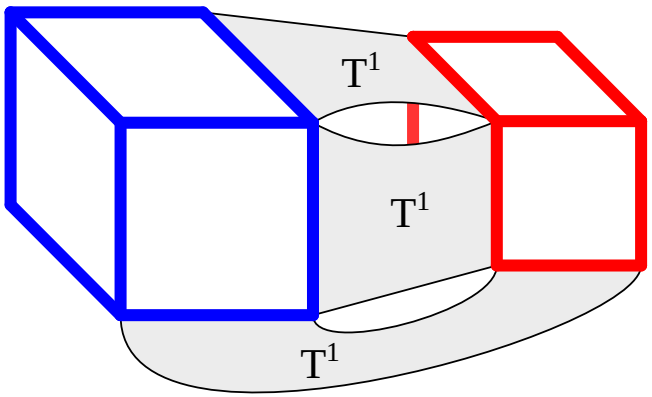
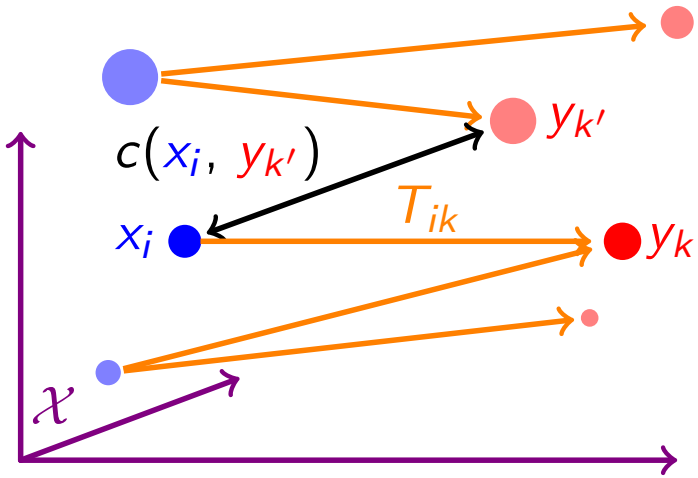


Table of Contents

Discrete Optimal Transport (OT) and Wasserstein distance



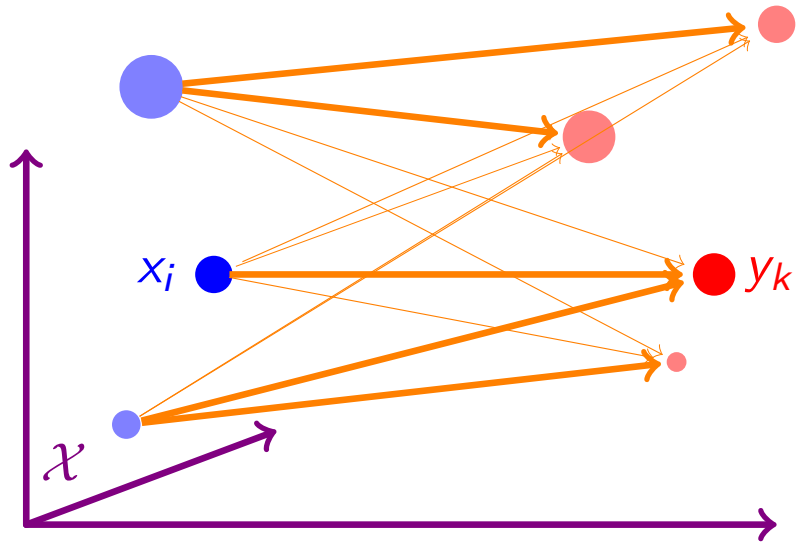
- The ground cost c is used to compare points.

Optimal Transport [Monge, 1781; Kantorovich, 1942; Villani, 2008]

- Two empirical distributions: $\mu = \sum_{i=1}^{I} a_i \delta_{x_i}$ and $\nu = \sum_{k=1}^{K} b_k \delta_{y_k}$.
- Set of transport plans: $\Pi(\mu,\nu) = \{T \in \mathbb{R}_+^{I \times K} \,|\, T^\top \mathbb{1}_I = b, T\mathbb{1}_K = a\}$

$$\min_{T \in \Pi(\mu,\nu)} \sum_{i,k=1}^{I,K} c(x_i, y_k) T_{ik} = \min_{T \in \Pi(\mu,\nu)} \langle C, T \rangle \tag{1}$$

Kullback-Leibler regularization

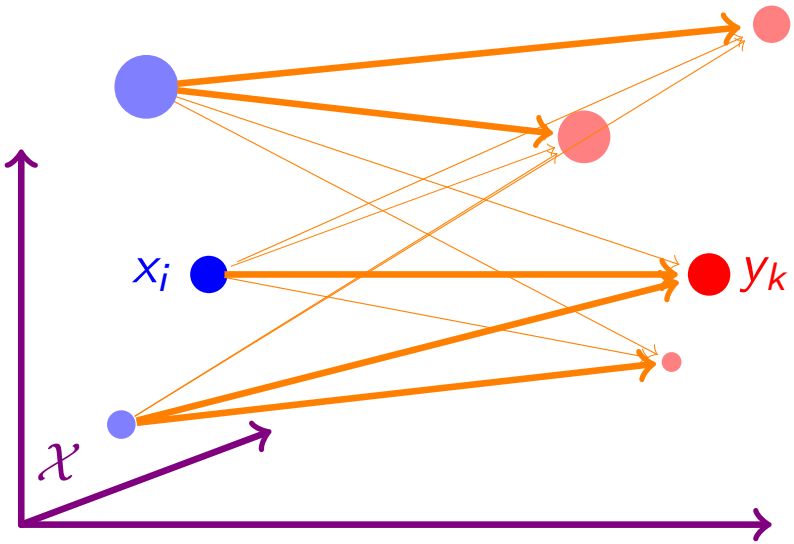


Entropy regularized OT [Cuturi, 2013; Sinkhorn and Knopp, 1967]

- Kullback-Leibler regularization $KL(T||T') = \sum_{i,k=1}^{I,K} T_{ik} log(\frac{T_{ik}}{T'_{ik}})$:

$$\min_{T \in \Pi(\mu,\nu)} \sum_{i,k=1}^{I,K} c(x_i, y_k) T_{ik} + \epsilon KL(T||ab^\top) \tag{2}$$

Kullback-Leibler regularization



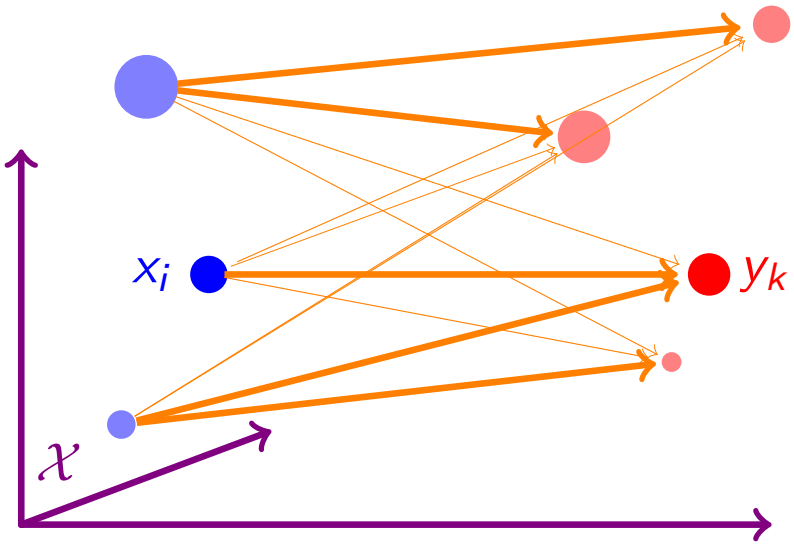
- The OT plan is not sparse anymore

Entropy regularized OT [Cuturi, 2013; Sinkhorn and Knopp, 1967]

- Kullback-Leibler regularization $KL(T||T') = \sum_{i,k=1}^{I,K} T_{ik} log(\frac{T_{ik}}{T'_{ik}})$:

$$\min_{T \in \Pi(\mu,\nu)} \sum_{i,k=1}^{I,K} c(x_i, y_k) T_{ik} + \epsilon KL(T||ab^\top) \tag{2}$$

Kullback-Leibler regularization



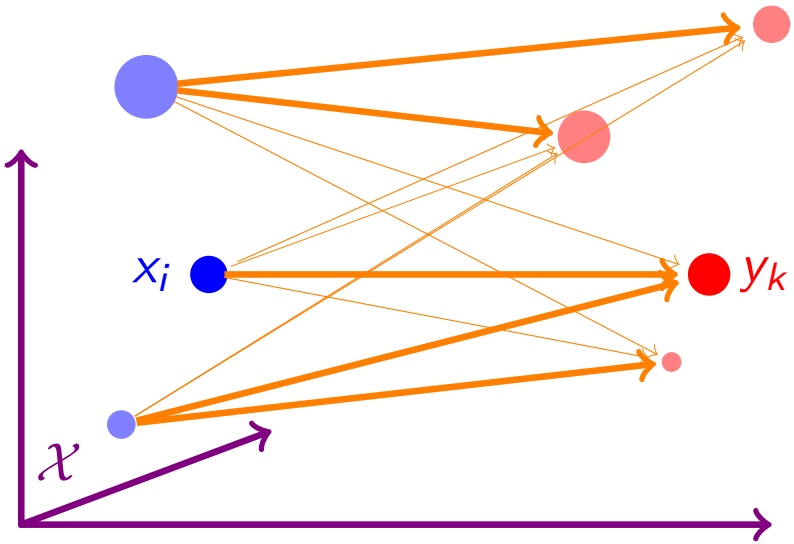
- The OT plan is not sparse anymore
- Unique solution

Entropy regularized OT [Cuturi, 2013; Sinkhorn and Knopp, 1967]

- Kullback-Leibler regularization $KL(T||T') = \sum_{i,k=1}^{I,K} T_{ik} log(\frac{T_{ik}}{T'_{ik}})$:

$$\min_{T \in \Pi(\mu,\nu)} \sum_{i,k=1}^{I,K} c(x_i, y_k) T_{ik} + \epsilon KL(T||ab^\top) \tag{2}$$

Kullback-Leibler regularization



- The OT plan is not sparse anymore
- Unique solution
- The OT plan changes continuously

Entropy regularized OT [Cuturi, 2013; Sinkhorn and Knopp, 1967]

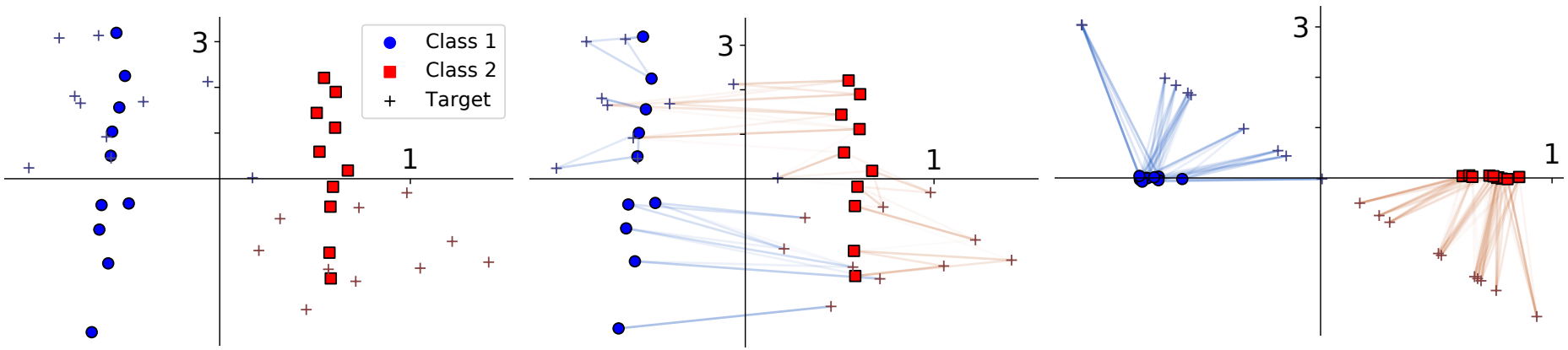
- Kullback-Leibler regularization $KL(T||T') = \sum_{i,k=1}^{I,K} T_{ik} log(\frac{T_{ik}}{T'_{ik}})$:

$$\min_{T \in \Pi(\mu,\nu)} \sum_{i,k=1}^{I,K} c(x_i, y_k) T_{ik} + \epsilon KL(T||ab^\top) \tag{2}$$

Table of Contents

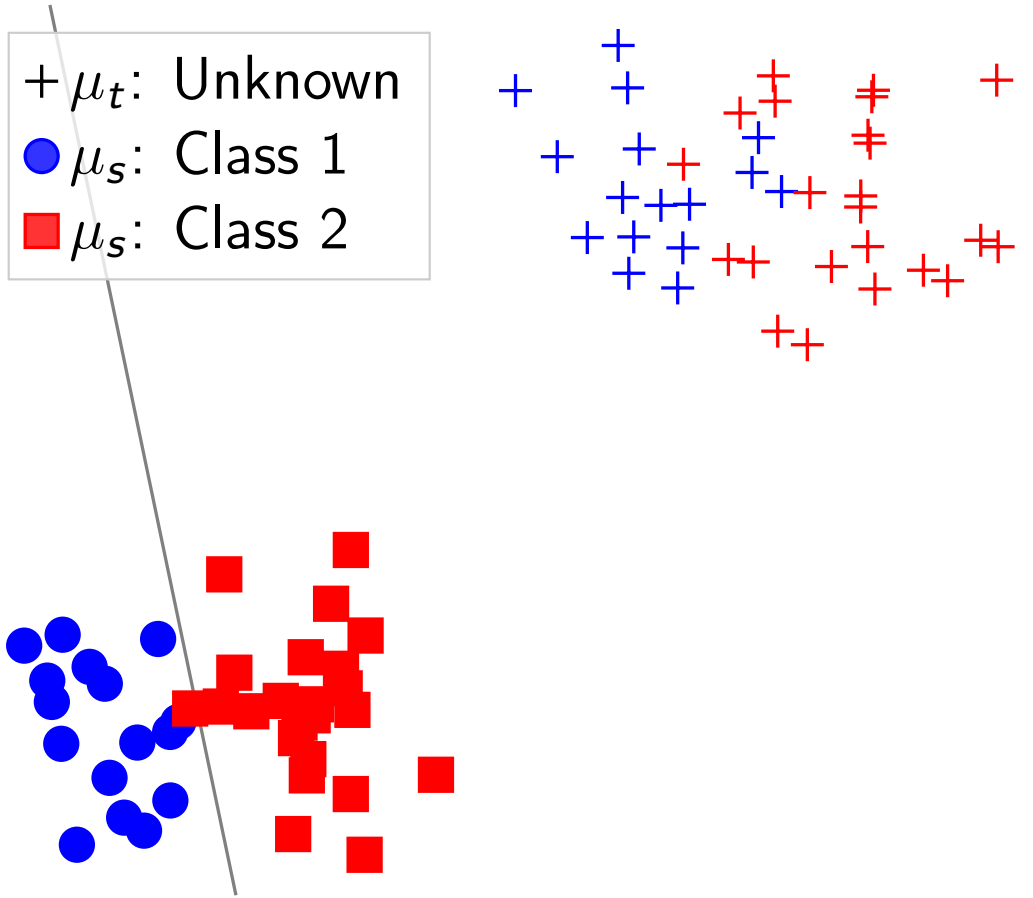
Metric Learning for Optimal Transport

Based on a published paper at the International Joint Conference on Artificial Intelligence (IJCAI) [Kerdoncuff et al., 2020]



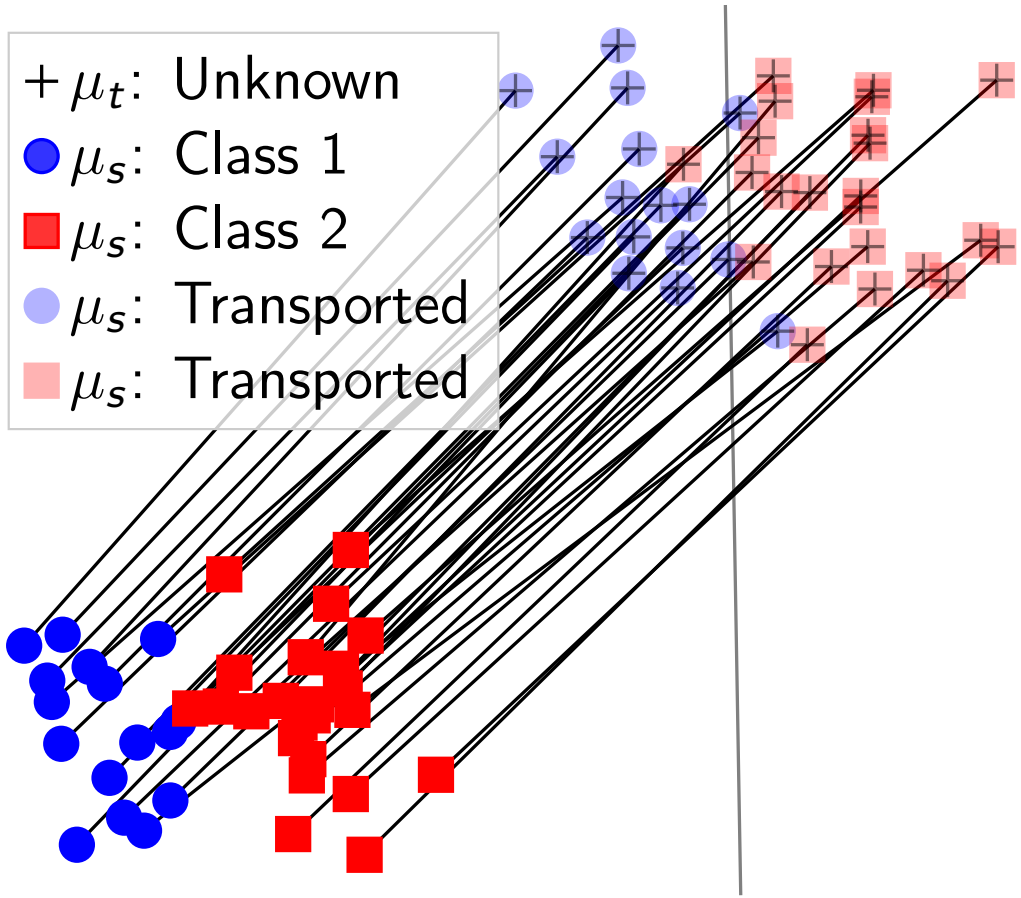
Intuition of Optimal Transport for Domain Adaptation

DIRECTION GÉNÉRALE DES
FINANCES PUBLIQUES

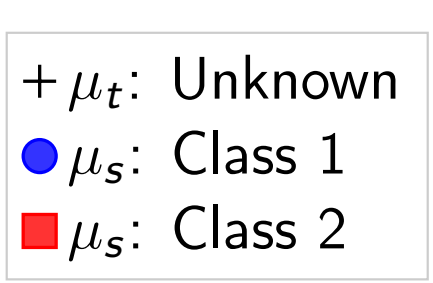


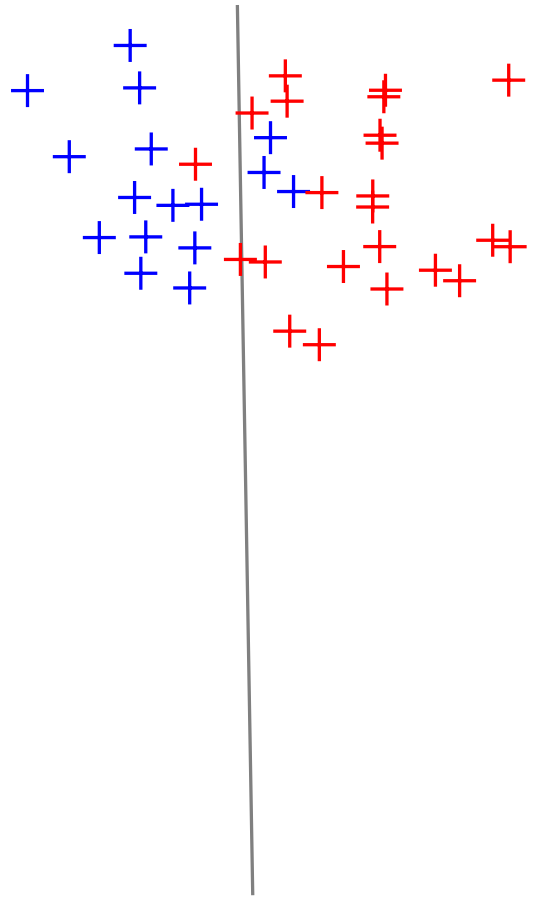
Intuition of Optimal Transport for Domain Adaptation

DIRECTION GÉNÉRALE DES
FINANCES PUBLIQUES

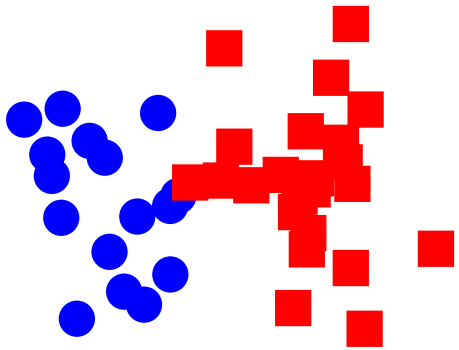


Intuition of Optimal Transport for Domain Adaptation





DIRECTION GÉNÉRALE DES
FINANCES PUBLIQUES



Optimal Transport for Domain Adaptation (OTDA)

Optimal Transport for Domain Adaptation [Courty et al., 2017]

$$\min_{T \in \Pi(\hat{\mu}_s, \hat{\mu}_t)} \langle T, C \rangle + \epsilon KL(T || ab^\top) + \lambda_c \underbrace{\left(\sum_{k=1}^{K} \sum_{cl=1}^{\#classes} \| T(\mathcal{I}_{cl}, k) \|_2 \right)}_{\Omega_c} \quad (3)$$

✓ Points of different classes should **not** be sent to the same location

Metric learning

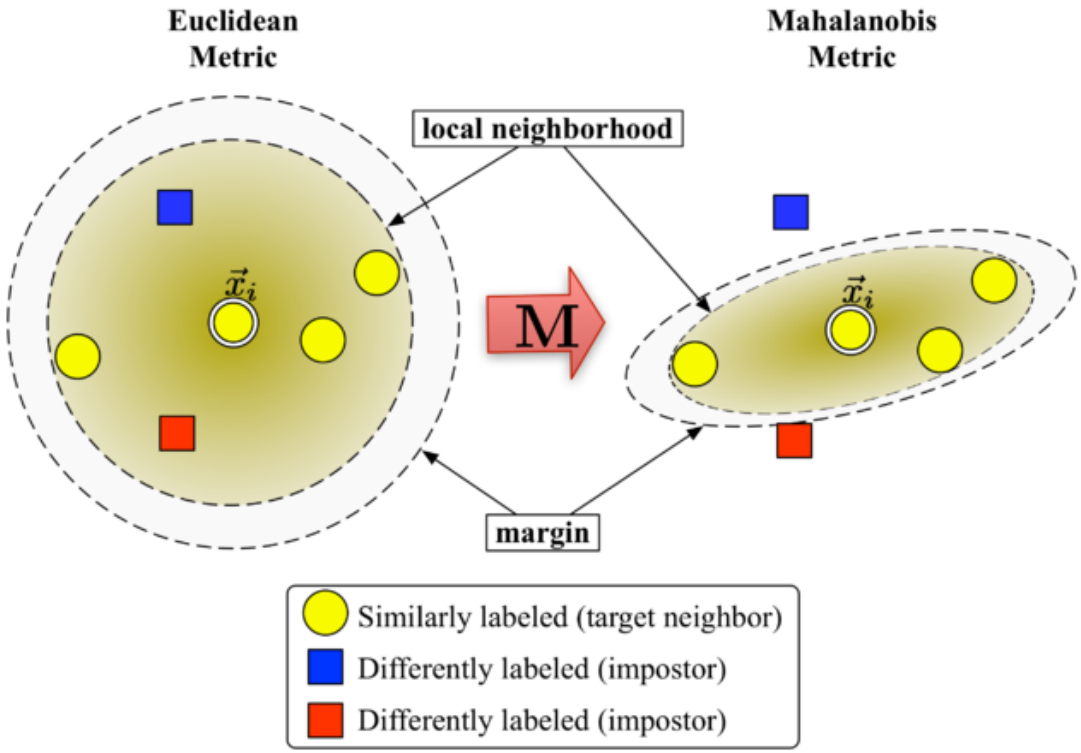


Figure 1: LMNN [Weinberger and Saul, 2009] CC-BY Mlguy (wikipedia)

Mahalanobis metric

- $C_M^2(x, x') = (x - x')^\top M(x - x') = \|Lx - Lx'\|_2^2$ with $L^\top L = M$

Metric Learning for Optimal Transport (MLOT)

Algorithm 1 MLOT

Require: η (gradient step) $\boldsymbol{X}^s$ $\boldsymbol{X}^t$ $\boldsymbol{Y}^s$

1: $\boldsymbol{V}_s = PCA(\boldsymbol{X}^s)$, $\boldsymbol{V}_t = PCA(\boldsymbol{X}^t)$

2: $\boldsymbol{L}_s = \boldsymbol{V}_s^\top \boldsymbol{V}_s$, $\boldsymbol{L}_t = \boldsymbol{V}_t^\top \boldsymbol{V}_t$

3: **for** $i = 1$ **to** P **do**

4: $\quad T = \underset{T \in \Pi(\hat{\mu}_s, \hat{\mu}_t)}{\operatorname{argmin}} \langle T, C^2(\boldsymbol{L}_s, \boldsymbol{L}_t) \rangle + \epsilon KL(T | ab^\top) + \lambda_c \Omega_{cl}(T)$

5: $\quad \boldsymbol{L}_s = \boldsymbol{L}_s - \eta \nabla_{\boldsymbol{L}_s} (\langle T, C^2(\boldsymbol{L}_s, \boldsymbol{L}_t) \rangle + \lambda_l \Omega_l(\boldsymbol{L}_s))$

6: **end for**

7: $\tilde{\boldsymbol{X}}^s = T \boldsymbol{L}_t \boldsymbol{X}^t$

8: classifier = learning classifier($\tilde{\boldsymbol{X}}^s$, $\boldsymbol{Y}^s$)

9: $\hat{\boldsymbol{Y}}^t$ = classifier($\boldsymbol{X}^t$)

10: **return** $\hat{\boldsymbol{Y}}^t$

OTDA vs MLOT

Figure 2: MLOT - OTDA

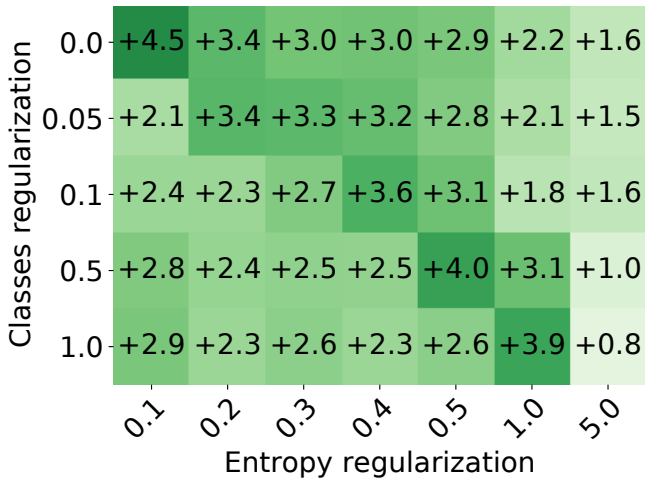
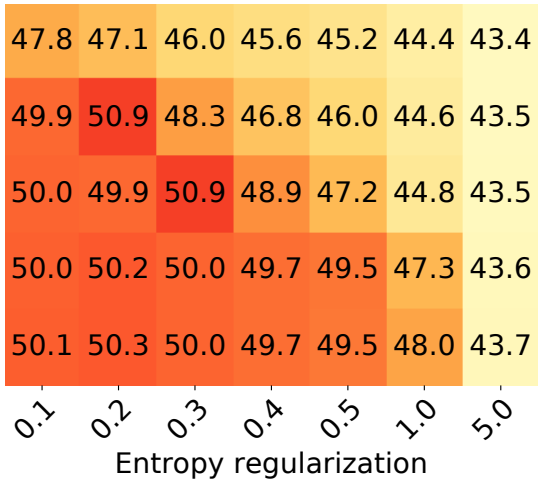


Figure 3: MLOT



OTDA and MLOT on Office-Caltech dataset [Gong et al., 2012] with SURF features

OTDA vs MLOT

Figure 2: MLOT - OTDA

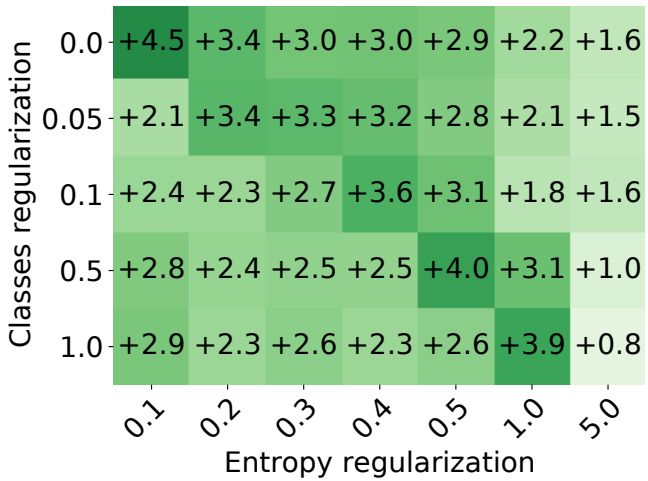
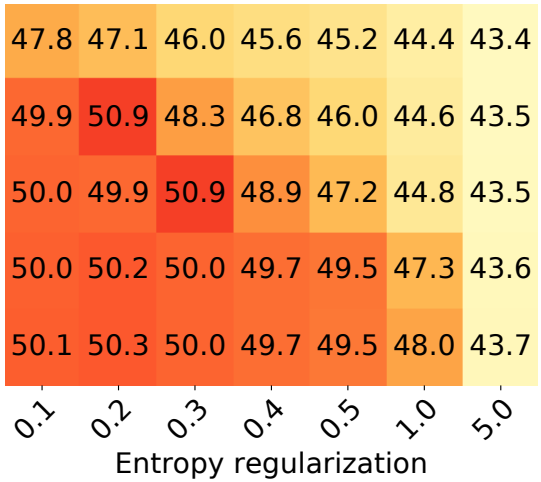


Figure 3: MLOT



OTDA and MLOT on Office-Caltech dataset [Gong et al., 2012] with SURF features

- It is always good to learn a metric

OTDA vs MLOT

Figure 2: MLOT - OTDA

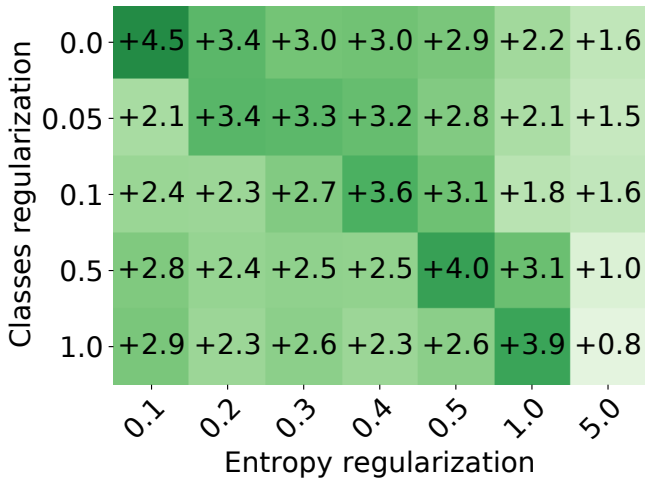
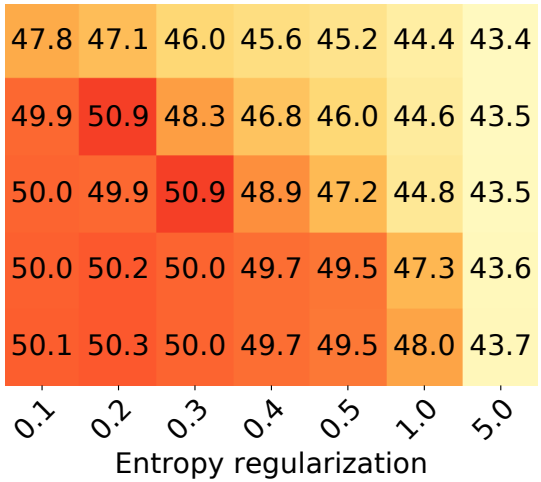


Figure 3: MLOT



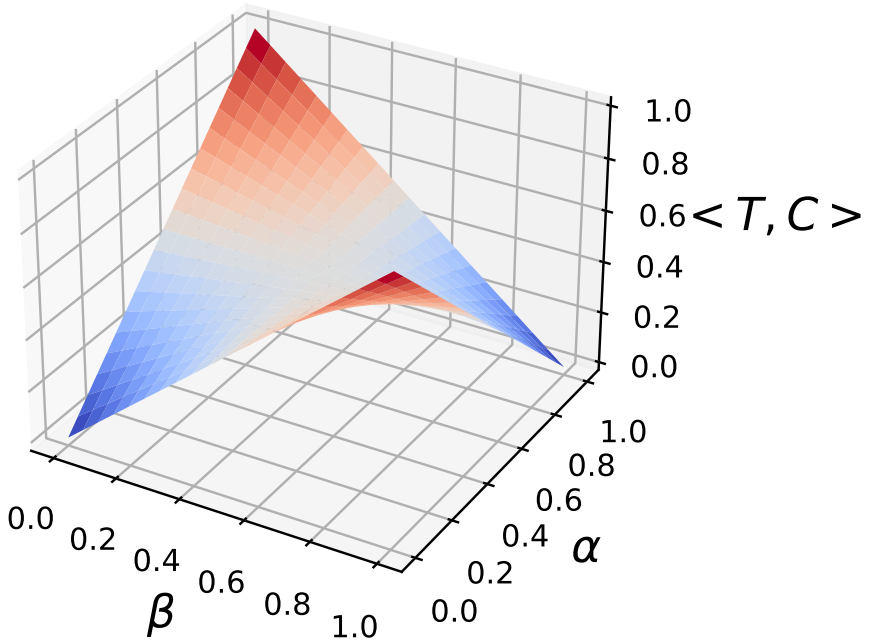
OTDA and MLOT on Office-Caltech dataset [Gong et al., 2012] with SURF features

- It is always good to learn a metric
- It is always good to use the OTDA class regularization

Table of Contents

A Swiss Army Knife for Minimax Optimal Transport

Based on a published paper at International Conference on Machine Learning (ICML) 2020 [Dhouib et al., 2020]



Different type of set

Robust Kantorovich Problem (RKP)

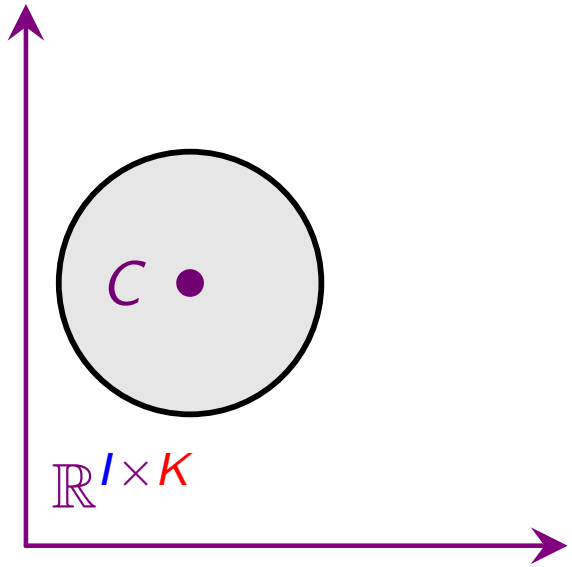
$$\mathcal{W}_{\mathcal{C}} = \min_{T \in \Pi} \max_{C \in \mathcal{C}} \langle T, C \rangle \tag{4}$$

Different type of set

Robust Kantorovich Problem (RKP)

$$\mathcal{W}_{\mathcal{C}} = \min_{T \in \Pi} \max_{C \in \mathcal{C}} \langle T, C \rangle \tag{4}$$

- Mahalanobis ball centered at a matrix C :

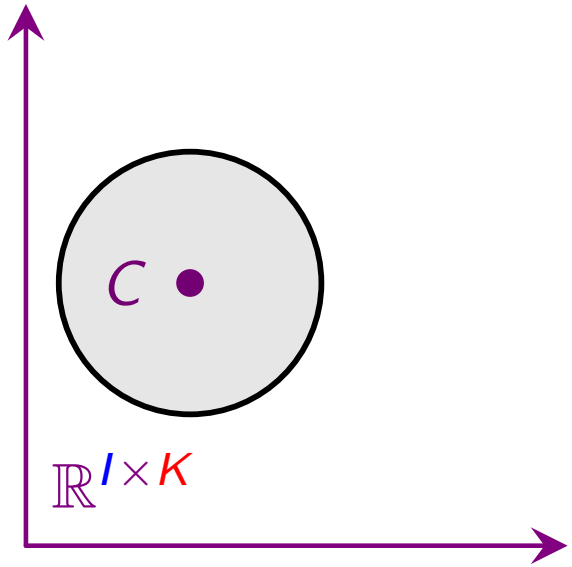


Different type of set

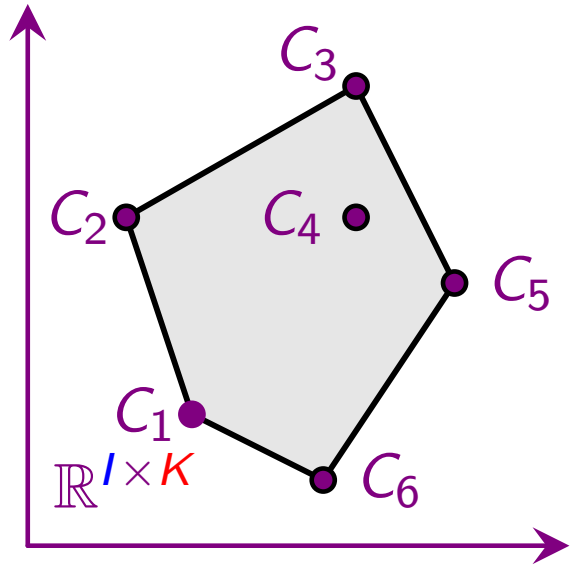
Robust Kantorovich Problem (RKP)

$$\mathcal{W}_{\mathcal{C}} = \min_{T \in \Pi} \max_{C \in \mathcal{C}} \langle T, C \rangle \tag{4}$$

- Mahalanobis ball centered at a matrix C :



- Convex hull of finite set of cost matrices:



Cutting set algorithm [Mutapcic and Boyd, 2009]

Algorithm 2 Cutting set method for the Robust Kantorovich Problem (RKP)

Input: $\mathcal{C}$, $\mathcal{P}_0 \subset \Pi$, thd
while $err >$ thd **do**
    Solve (5) to obtain (ω_t, C_t) and $(q_0, ..., q_{|\mathcal{P}_t|-1})$
    Find $T_t \in \underset{T \in \Pi}{\operatorname{argmin}} \langle T, C_t \rangle$
    $err \leftarrow (\omega_t - \langle T_t, C_t \rangle) / \langle T_t, C_t \rangle$
    $\mathcal{P}_{t+1} = \mathcal{P}_t \cup \{T_t\}$
    $t \leftarrow t + 1$
end while
return $\sum_{l=0}^{|\mathcal{P}_t|-1} q_l T_l,\ C_t$

Cutting set algorithm [Mutapcic and Boyd, 2009]

Algorithm 3 Cutting set method for the Robust Kantorovich Problem (RKP)

Input: $\mathcal{C}$, $\mathcal{P}_0 \subset \Pi$, thd
while $err >$ thd **do**
    Solve (5) to obtain (ω_t, C_t) and $(q_0, ..., q_{|\mathcal{P}_t|-1})$
    Find $T_t \in \underset{T \in \Pi}{\operatorname{argmin}} \langle T, C_t \rangle$
    $err \leftarrow (\omega_t - \langle T_t, C_t \rangle) / \langle T_t, C_t \rangle$
    $\mathcal{P}_{t+1} = \mathcal{P}_t \cup \{T_t\}$
    $t \leftarrow t + 1$
end while
return $\sum_{l=0}^{|\mathcal{P}_t|-1} q_l T_l,\ C_t$

Robust Kantorovich Problem (RKP)

$$\mathcal{W}_{\mathcal{C}} = \min_{T \in Conv(\mathcal{P})} \max_{C \in \mathcal{C}} \langle T, C \rangle$$

Cutting set algorithm [Mutapcic and Boyd, 2009]

Algorithm 4 Cutting set method for the Robust Kantorovich Problem (RKP)

1: **Input:** $\mathcal{C}$, $\mathcal{P}_0 \subset \Pi$, thd
2: **while** $err >$ thd **do**
3: Solve (5) to obtain (ω_t, C_t) and $(q_0, ..., q_{|\mathcal{P}_t|-1})$
4: Find $T_t \in \underset{T \in \Pi}{\operatorname{argmin}} \langle T, C_t \rangle$
5: $err \leftarrow (\omega_t - \langle T_t, C_t \rangle) / \langle T_t, C_t \rangle$
6: $\mathcal{P}_{t+1} = \mathcal{P}_t \cup \{T_t\}$
7: $t \leftarrow t + 1$
8: **end while**
9: **return** $\sum_{l=0}^{|\mathcal{P}_t|-1} q_l T_l$, C_t

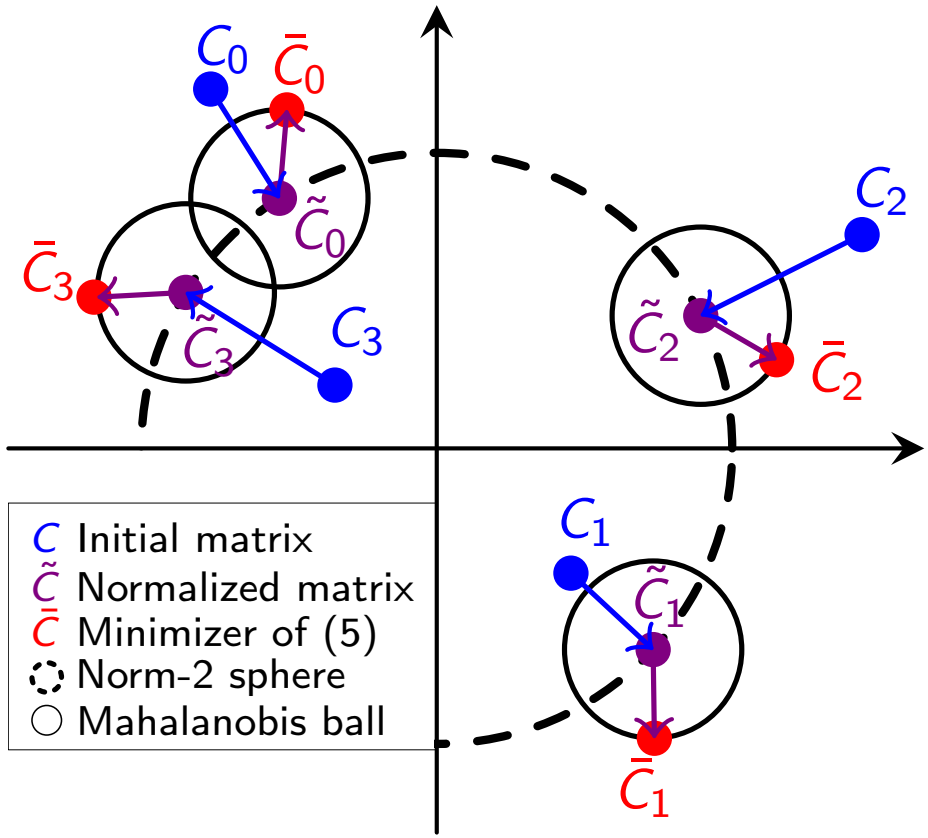
Robust Kantorovich Problem (RKP)

$$\mathcal{W}_{\mathcal{C}} = \min_{T \in Conv(\mathcal{P})} \max_{C \in \mathcal{C}} \langle T, C \rangle \quad \Longleftrightarrow \quad \begin{array}{l} \underset{C \in \mathcal{C}, \omega \geq 0}{\operatorname{argmax}} \ \omega \\ \text{s.t.} \ \langle T, C \rangle \geq \omega, \ \forall T \in \mathcal{P} \end{array} \tag{5}$$

Wasserstein stability

Wasserstein stability of a matrix C induced by μ and ν

$$\mathcal{WS}_C = \mathcal{W}_{\mathcal{C}_C}(\mu, \nu) - \mathcal{W}_C(\mu, \nu)$$



Usefulness of the Wasserstein stability (Color transfer)

Usefulness of the Wasserstein stability (Color transfer)

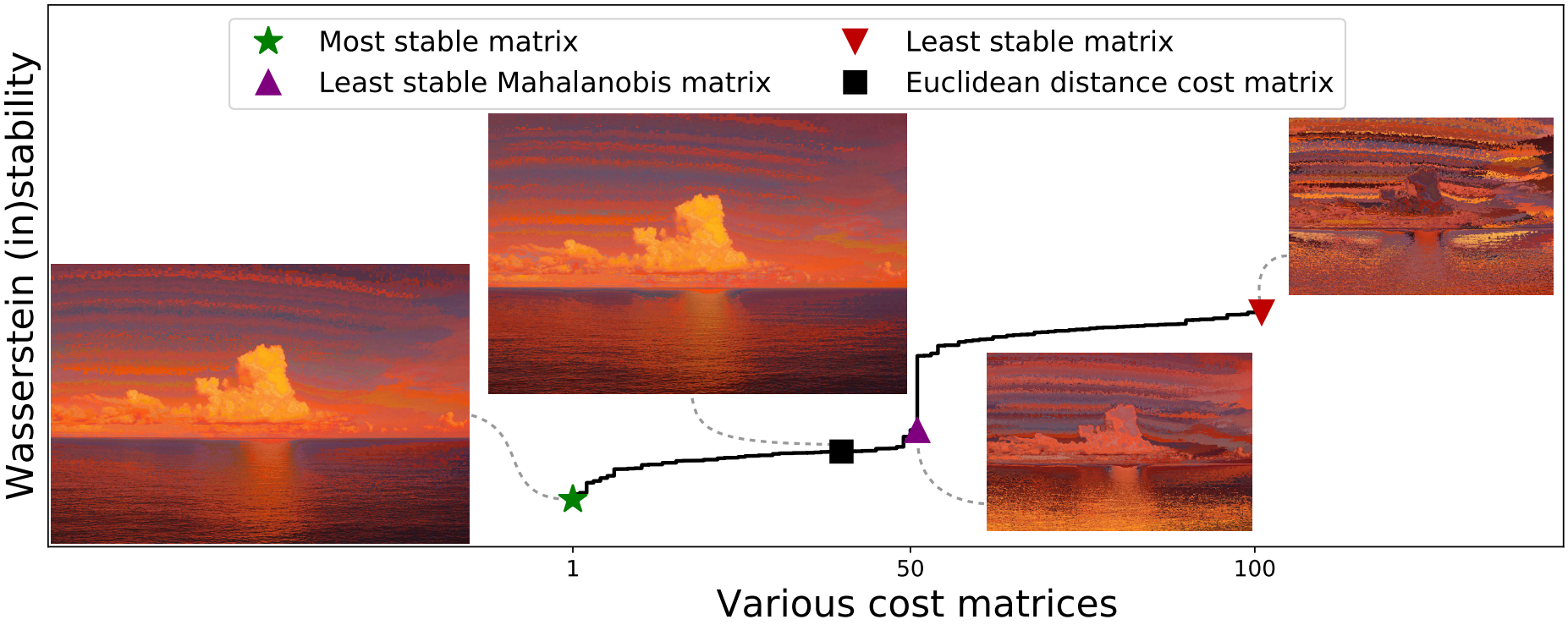
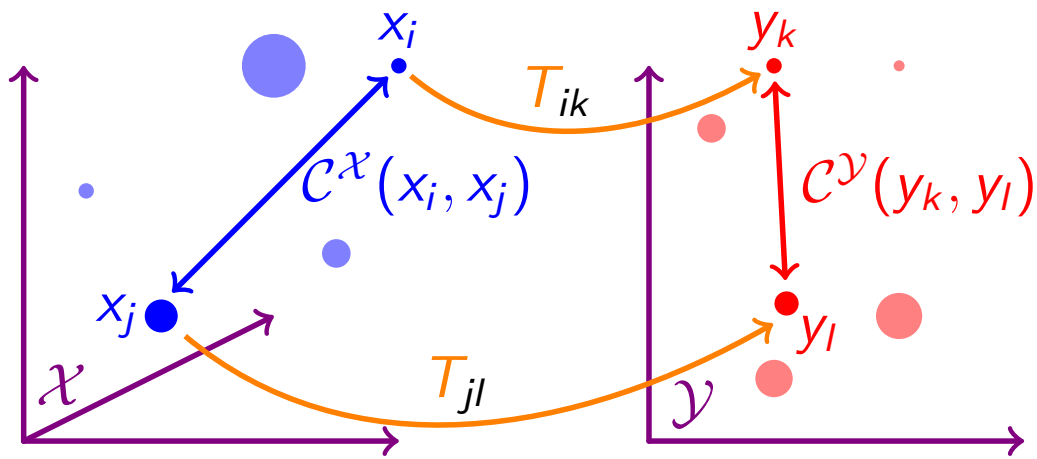


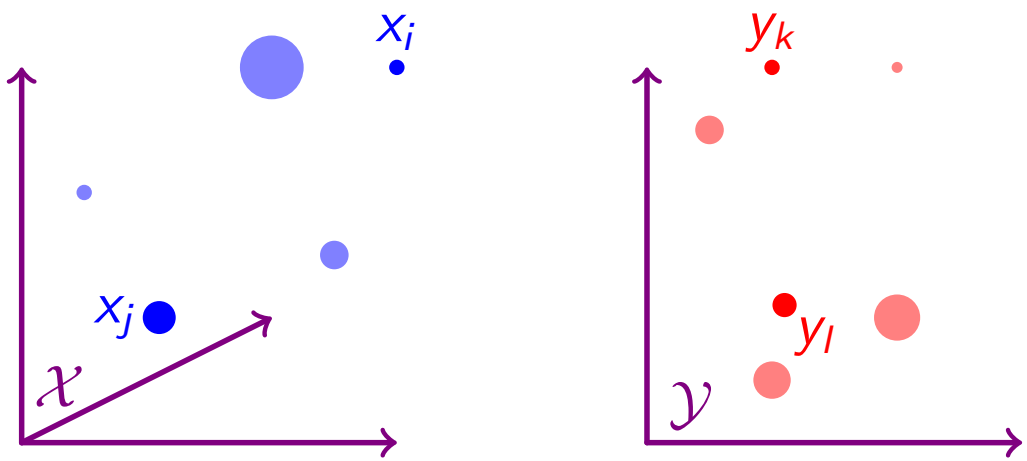
Table of Contents

Sampled Gromov Wasserstein

Based on a published paper in the Machine Learning Journal (MLJ) [Kerdoncuff et al., 2021] and presented at the ECML-PKDD 2021 conference.



The Gromov Wasserstein problem

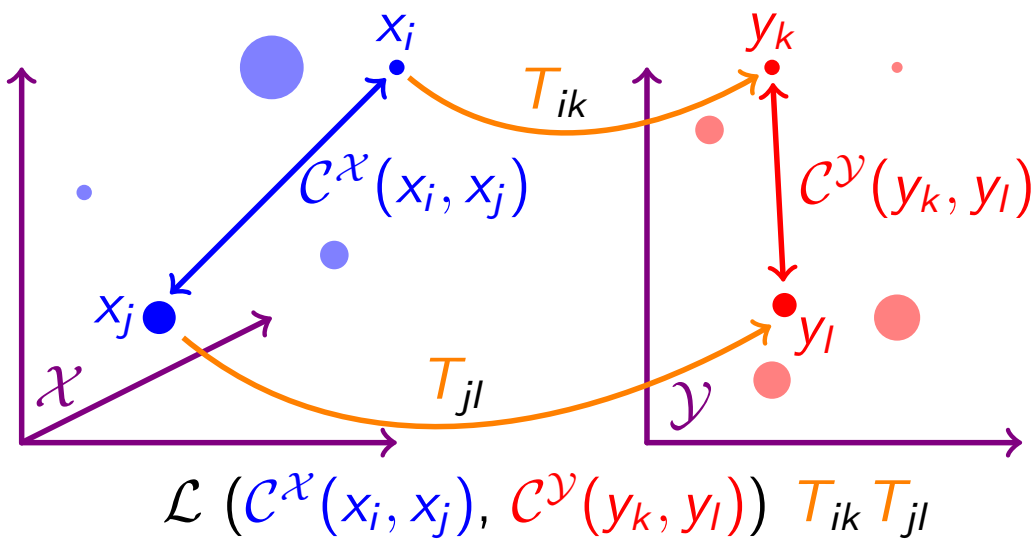


Gromov Wasserstein (GW) [Memoli, 2007; Peyré et al., 2016]

- The GW distance relies only on intra-pairwise distances,

$$\min_{T \in \Pi(\mu,\nu)} \mathcal{E}(T, T) = \min_{T \in \Pi(\mu,\nu)} \sum_{ijkl=1}^{I,I,K,K} \mathcal{L}\left(\mathcal{C}^{\mathcal{X}}(x_i, x_j), \mathcal{C}^{\mathcal{Y}}(y_k, y_l)\right) T_{ik} T_{jl} \quad (6)$$

The Gromov Wasserstein problem



Gromov Wasserstein (GW) [Memoli, 2007; Peyré et al., 2016]

- The GW distance relies only on intra-pairwise distances,

$$\min_{T \in \Pi(\mu,\nu)} \mathcal{E}(T, T) = \min_{T \in \Pi(\mu,\nu)} \sum_{ijkl=1}^{I,I,K,K} \mathcal{L}\left(\mathcal{C}^{\mathcal{X}}(x_i, x_j), \mathcal{C}^{\mathcal{Y}}(y_k, y_l)\right) T_{ik} T_{jl} \quad (6)$$

The most used solver: Entropy Gromov Wasserstein

Algorithm 5 Entropy Gromov Wasserstein (EGW)
[Peyré et al., 2016; Rangarajan et al., 1999]

Require: $a, b, \mathcal{C}^{\mathcal{X}}, \mathcal{C}^{\mathcal{Y}}, \mathcal{L}, \epsilon$

1: $T_0 = ab^\top$

2: **for** s= 0 **to** S-1 **do**

3: $\Lambda = \nabla_{T_s} \mathcal{E}(T_s, T_s) = \sum_{j,l=1}^{I,K} \mathcal{L}\left(\mathcal{C}^{\mathcal{X}}(\cdot, x_j), \mathcal{C}^{\mathcal{Y}}(\cdot, y_l)\right)(T_s)_{jl}$

4: $T_s = \min_{T' \in \Pi(\mu,\nu)} \langle \Lambda, T' \rangle + \epsilon KL(T' || ab^\top)$

5: **end for**

Motivations and contributions

Limitation of EGW

- Time complexity for computing the gradient $\nabla_T \mathcal{E}(T, T)$ is $O\left(N^4\right)$
- Time complexity can be reduced to $O\left(N^3\right)$ but only for specific loss functions $\mathcal{L}(x, y) = f_1(x) + h_1(x)h_2(y) + f_2(y)$

Motivations and contributions

Limitation of EGW

- Time complexity for computing the gradient $\nabla_T \mathcal{E}(T, T)$ is $O\left(N^4\right)$
- Time complexity can be reduced to $O\left(N^3\right)$ but only for specific loss functions $\mathcal{L}(x, y) = f_1(x) + h_1(x)h_2(y) + f_2(y)$

Contributions

- ✓ Approximate the gradient $\nabla_T \mathcal{E}(T, T)$ in $O\left(M \times N^2\right)$ for any loss
- ✓ Propose a convergence bound to stationary points
- ✓ Explore a very fast variant with $M = 1$ using the 1D OT solver

Sampled Gromov Wasserstein idea (SaGroW)

$$\min_{T' \in \Pi(\mu,\nu)} \left\langle \nabla_T \mathcal{E}(T, T), T' \right\rangle = \min_{T' \in \Pi(\mu,\nu)} \left\langle \sum_{j,l=1}^{I,K} \mathcal{L}\left(\mathcal{C}^{\mathcal{X}}(\cdot, x_j), \mathcal{C}^{\mathcal{Y}}(\cdot, y_l)\right) T_{jl}, T' \right\rangle$$

Sampled Gromov Wasserstein idea (SaGroW)

$$\min_{T' \in \Pi(\mu,\nu)} \left\langle \nabla_T \mathcal{E}(T, T), T' \right\rangle = \min_{T' \in \Pi(\mu,\nu)} \left\langle \sum_{j,l=1}^{I,K} \mathcal{L}\left(\mathcal{C}^{\mathcal{X}}(\cdot, x_j), \mathcal{C}^{\mathcal{Y}}(\cdot, y_l)\right) T_{jl}, T' \right\rangle$$

$$= \min_{T' \in \Pi(\mu,\nu)} \left\langle \sum_{j,l=1}^{I,K} \boldsymbol{L}_{.j.l}\, T_{jl}, T' \right\rangle$$

Sampled Gromov Wasserstein idea (SaGroW)

$$\min_{T' \in \Pi(\mu,\nu)} \langle \nabla_T \mathcal{E}(T, T), T' \rangle = \min_{T' \in \Pi(\mu,\nu)} \left\langle \sum_{j,l=1}^{I,K} \mathcal{L}\left(\mathcal{C}^{\mathcal{X}}(\cdot, x_j), \mathcal{C}^{\mathcal{Y}}(\cdot, y_l)\right) T_{jl}, T' \right\rangle$$

$$= \min_{T' \in \Pi(\mu,\nu)} \left\langle \sum_{j,l=1}^{I,K} \boldsymbol{L}_{.j.l}\, T_{jl}, T' \right\rangle$$

$$= \min_{T' \in \Pi(\mu,\nu)} \langle \mathbb{E}(\Lambda), T' \rangle$$

Sampled Gromov Wasserstein idea (SaGroW)

$$\min_{T' \in \Pi(\mu,\nu)} \langle \nabla_T \mathcal{E}(T, T), T' \rangle = \min_{T' \in \Pi(\mu,\nu)} \left\langle \sum_{j,l=1}^{I,K} \mathcal{L}\left(\mathcal{C}^{\mathcal{X}}(\cdot, x_j), \mathcal{C}^{\mathcal{Y}}(\cdot, y_l)\right) T_{jl}, T' \right\rangle$$

$$= \min_{T' \in \Pi(\mu,\nu)} \left\langle \sum_{j,l=1}^{I,K} \boldsymbol{L}_{.j.l}\, T_{jl}, T' \right\rangle$$

$$= \min_{T' \in \Pi(\mu,\nu)} \langle \mathbb{E}(\Lambda), T' \rangle$$

$$\approx \min_{T' \in \Pi(\mu,\nu)} \left\langle \frac{1}{M} \sum_{m=1}^{M} \boldsymbol{C}^m, T' \right\rangle$$

SaGroW algorithm

- SaGrow has been implemented in POT [Flamary and Courty, 2017]

Algorithm 6 SaGroW

Require: $a, b, \mathcal{C}^{\mathcal{X}}, \mathcal{C}^{\mathcal{Y}}, \mathcal{L}, M, \epsilon, \alpha$

1: $T_0 = ab^\top$

2: **for** s= 0 **to** S-1 **do**

3: $\quad (j_m, l_m) \sim \text{Sample}(T_s)\ \forall m \in [\![1, M]\!]$

4: $\quad \widehat{\mathbf{\Lambda}} = \frac{1}{M} \sum_{m=1}^{M} \mathcal{L}\left(\mathcal{C}^{\mathcal{X}}(\cdot, x_{j_m}), \mathcal{C}^{\mathcal{Y}}(\cdot, y_{l_m})\right)$

5: $\quad T'_s = \min_{T' \in \Pi(\mu,\nu)} \left\langle \widehat{\mathbf{\Lambda}}, T' \right\rangle + \epsilon KL(T' || ab^\top)$

6: $\quad T_{s+1} = (1-\alpha) T_s + \alpha T'_s$

7: **end for**

Bound for convergence to a stationary point

- The FW gap: $G(T) = \mathcal{E}(T, T) - \min_{T' \in \Pi(\mu,\nu)} \mathcal{E}(T, T')$.
- $G(T) = 0 \iff T$ is a stationary point of GW.

Bound for convergence to a stationary point

- The FW gap: $G(T) = \mathcal{E}(T, T) - \min_{T' \in \Pi(\mu,\nu)} \mathcal{E}(T, T')$.
- $G(T) = 0 \iff T$ is a stationary point of GW.

Theorem (Based on Reddi et al. [2016])

For any $\boldsymbol{L}_{ijkl} \in [0, B]$, for any distributions μ and ν with uniform weights a and b respectively, for any optimal solution T^* of Problem (6), on average for the transport plan $\overline{T}$ uniformly sampled from $(T_s)_{s \in [\![0, S-1]\!]}$, on average over all the samplings, the following bound holds:

$$\mathbb{E}\left(G\left(\overline{T}\right)\right) \leq \sqrt{\frac{2B\left(\mathcal{E}(T_0) - \mathcal{E}(T^*)\right)N}{S}} + B\sqrt{\frac{2N}{M}} + \epsilon \log(N).$$

Stochastic Mirror Descent

Kullback Leibler regularization [Xu et al., 2019]

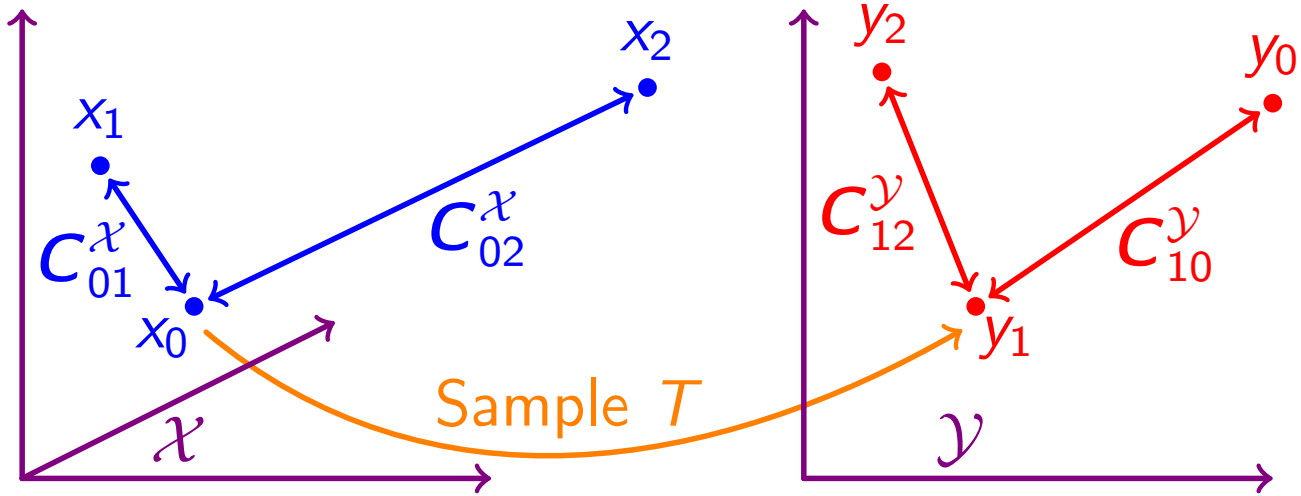
$$\left\langle \widehat{\mathbf{\Lambda}}, T' \right\rangle + \epsilon KL(T' || T) = \left\langle \widehat{\mathbf{\Lambda}} - \epsilon log(T), T' \right\rangle + \epsilon \sum_{ik} T'_{ik} log(T'_{ik}) \quad (7)$$

Algorithm 7 SaGroW with KL regularization

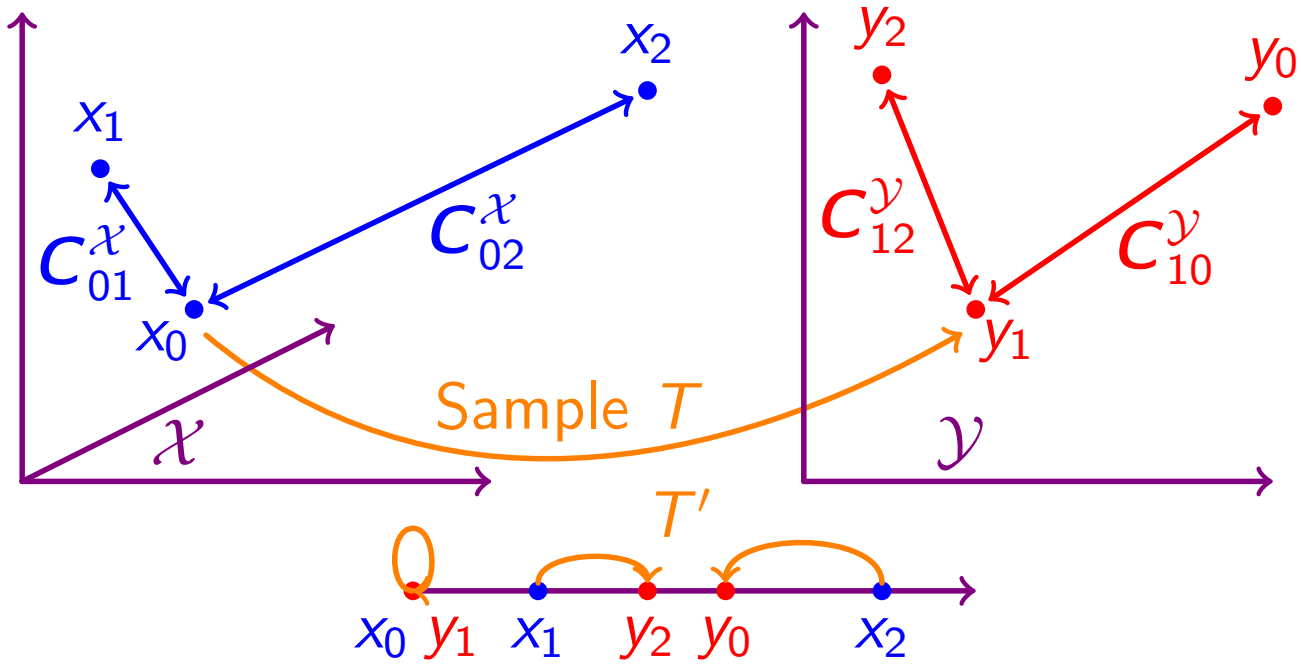
Require: $a, b, \mathcal{C}^{\mathcal{X}}, \mathcal{C}^{\mathcal{Y}}, \mathcal{L}, M, \epsilon, \alpha$

1: $T_0 = ab^\top$
2: **for** s= 0 **to** S-1 **do**
3: $(j_m, l_m) \sim \text{Sample}(T_s)\ \forall m \in [\![1, M]\!]$
4: $\widehat{\mathbf{\Lambda}} = \frac{1}{M} \sum_{m=1}^{M} \mathcal{L}\left(\mathcal{C}^{\mathcal{X}}(\cdot, x_{j_m}), \mathcal{C}^{\mathcal{Y}}(\cdot, y_{l_m})\right)$
5: $T_{s+1} = \min_{T' \in \Pi(\mu,\nu)} \left\langle \widehat{\mathbf{\Lambda}} - \epsilon log(T_s), T' \right\rangle - \epsilon \mathcal{H}(T')$
6: **end for**

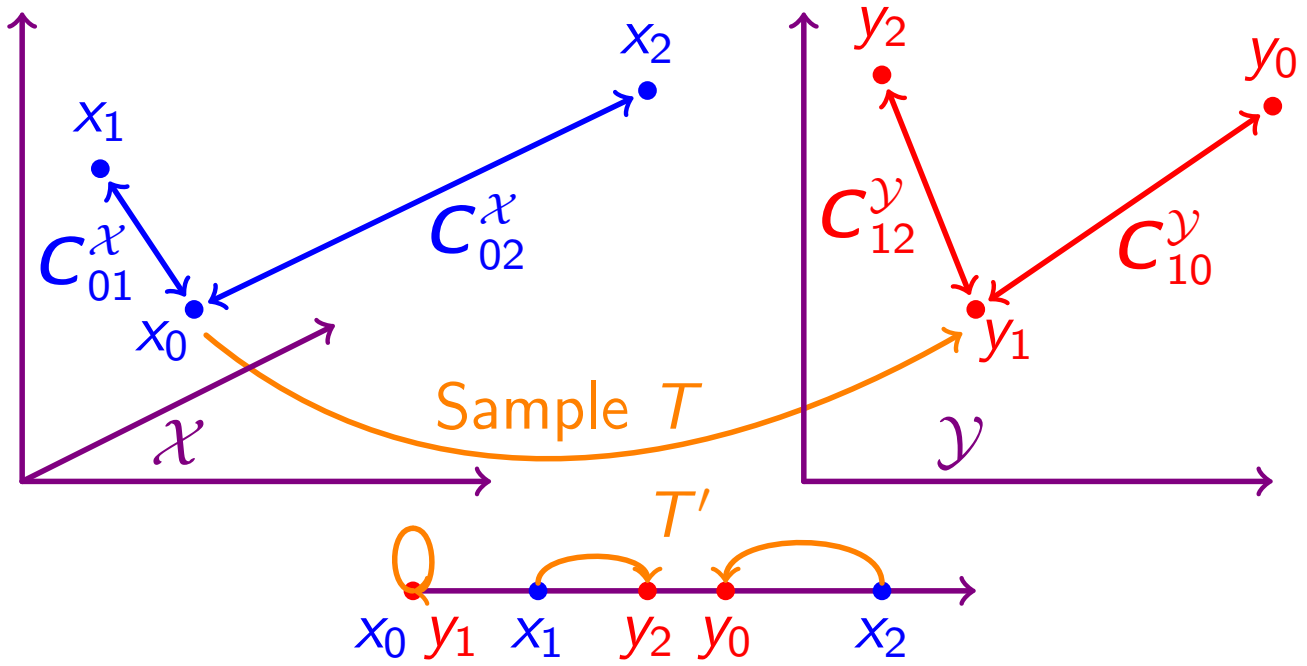
Pointwise Gromov Wasserstein (PoGroW): $M = 1$



Pointwise Gromov Wasserstein (PoGroW): $M = 1$



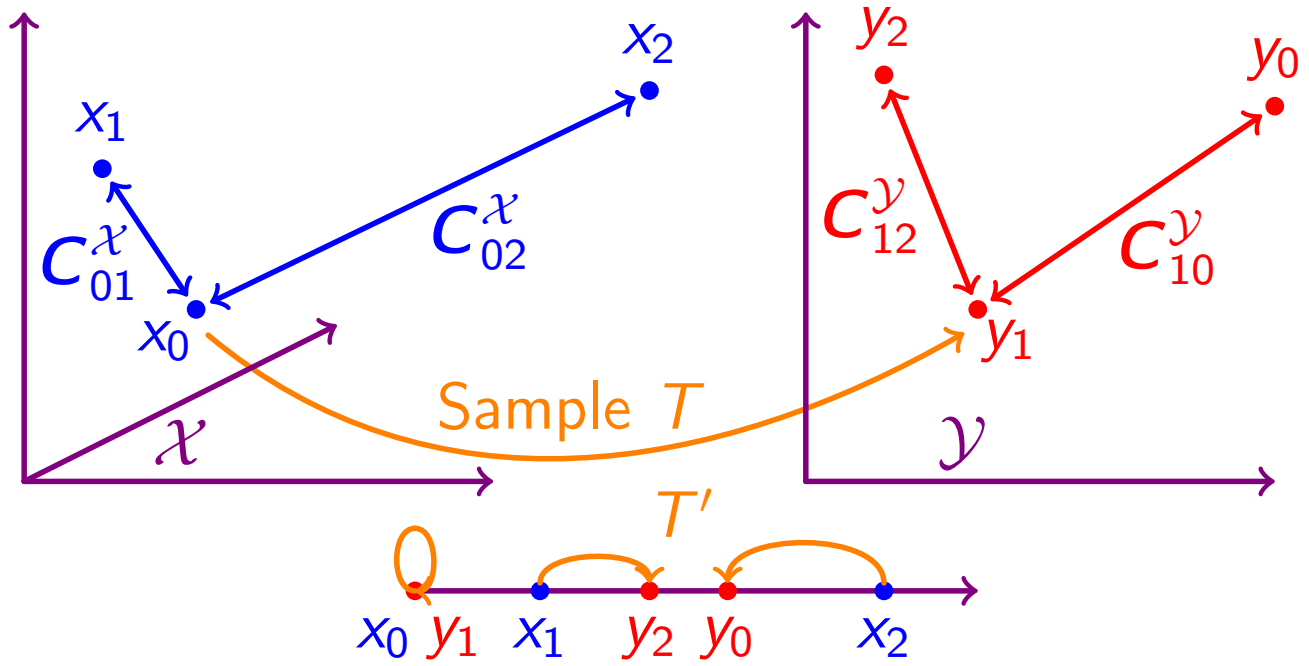
Pointwise Gromov Wasserstein (PoGroW): $M = 1$



Sliced Gromov Wasserstein (SGW) [Vayer et al., 2019]

- Approximate a distance related to the GW distance

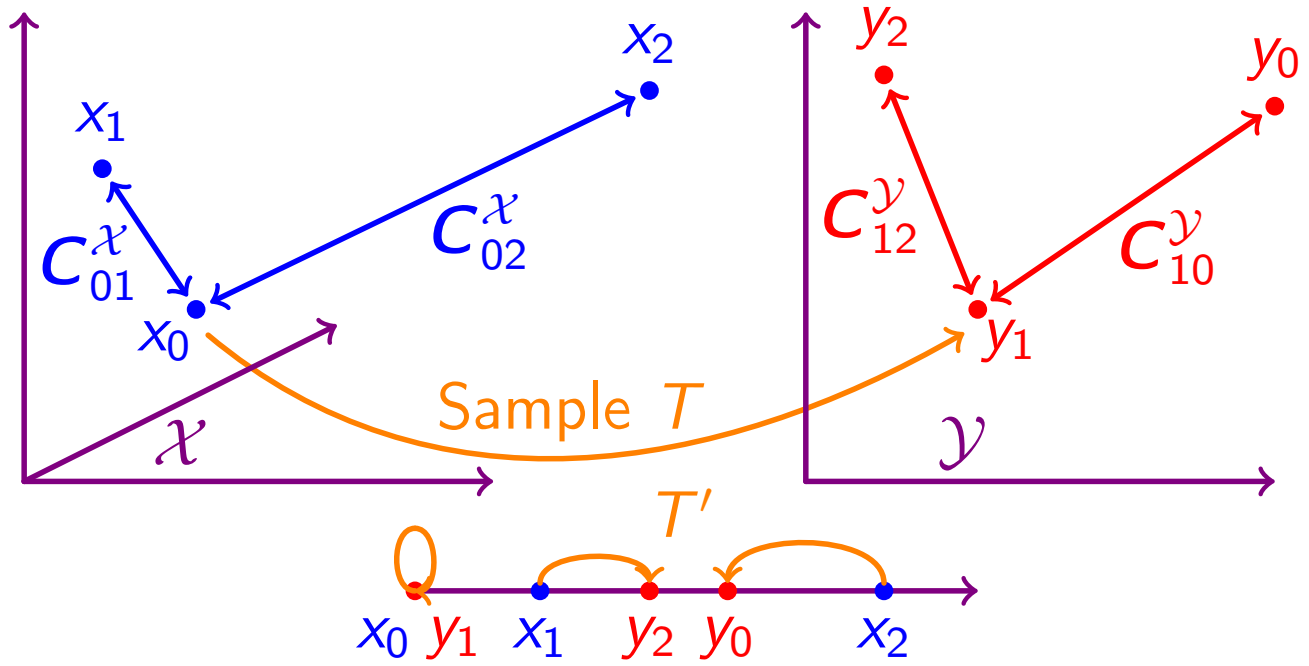
Pointwise Gromov Wasserstein (PoGroW): $M = 1$



Sliced Gromov Wasserstein (SGW) [Vayer et al., 2019]

- Approximate a distance related to the GW distance
- Does not provide a transport plan

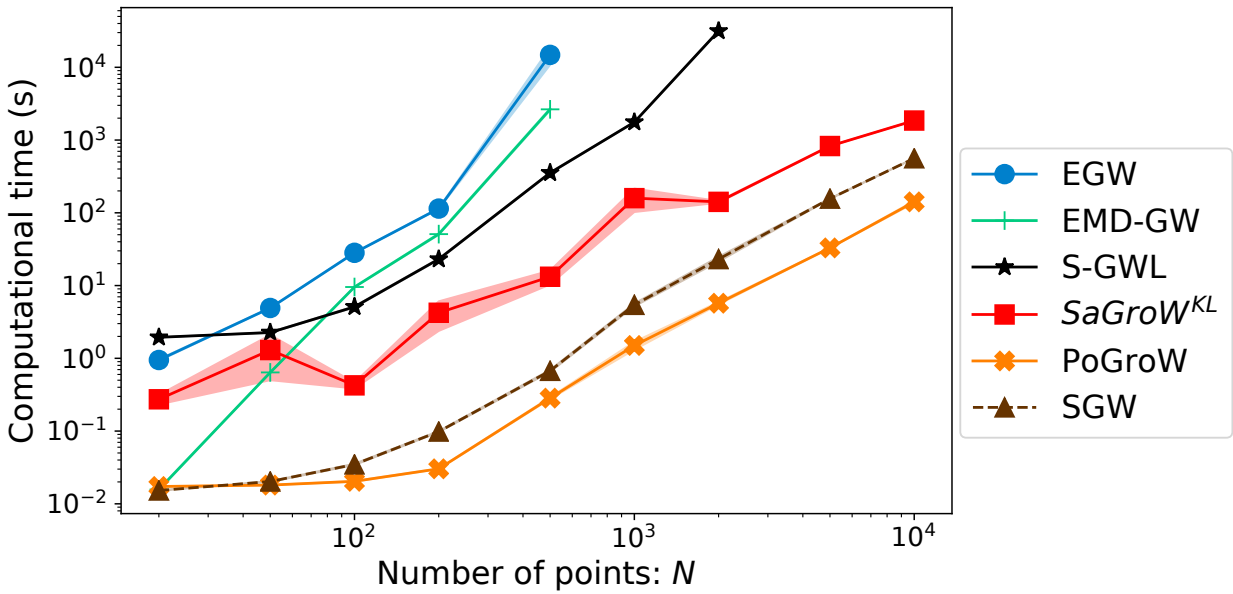
Pointwise Gromov Wasserstein (PoGroW): $M = 1$



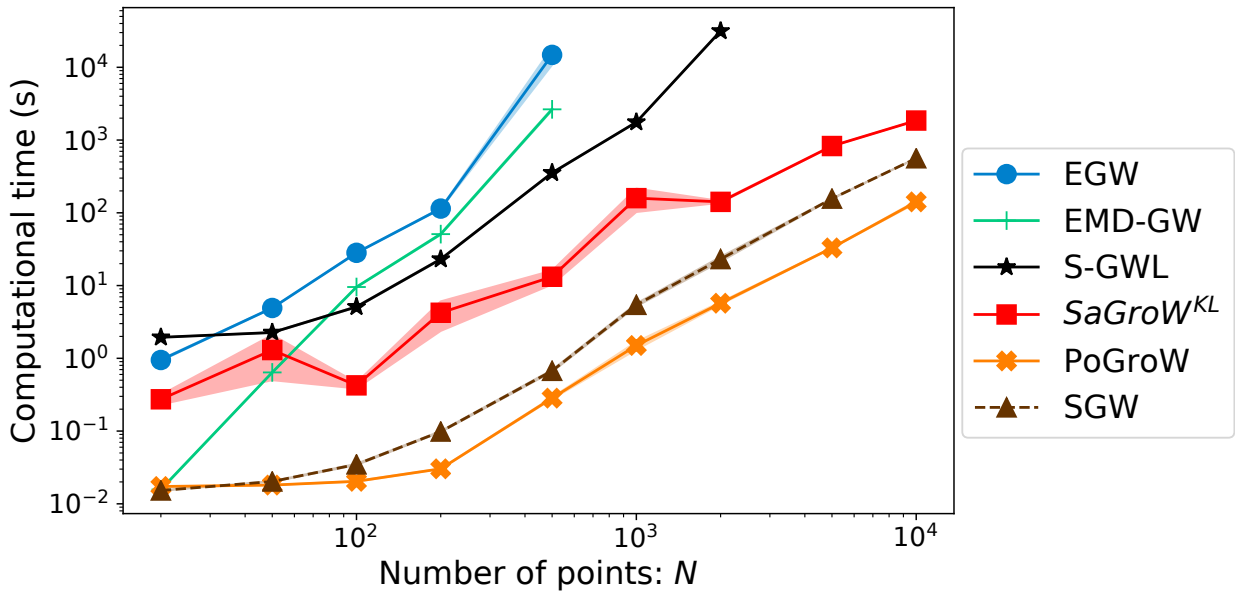
Sliced Gromov Wasserstein (SGW) [Vayer et al., 2019]

- Approximate a distance related to the GW distance
- Does not provide a transport plan
- Does not work on graphs

Speed and performances



Speed and performances



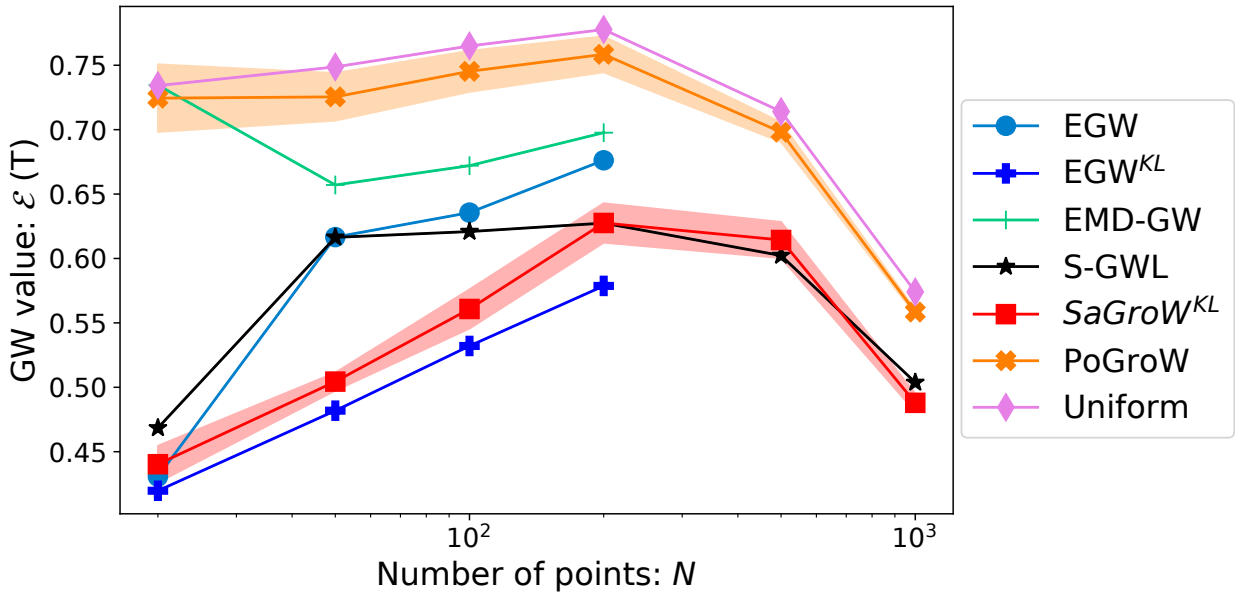
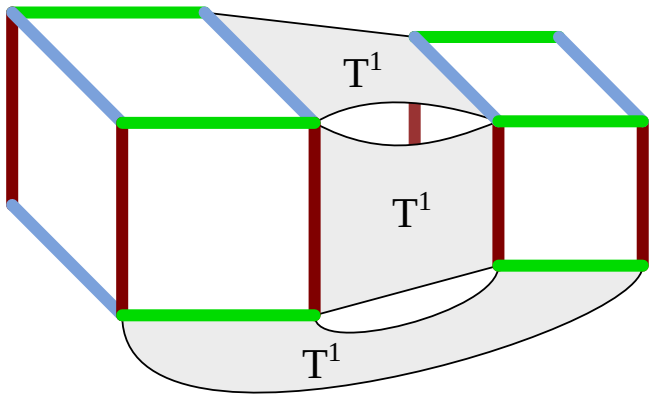


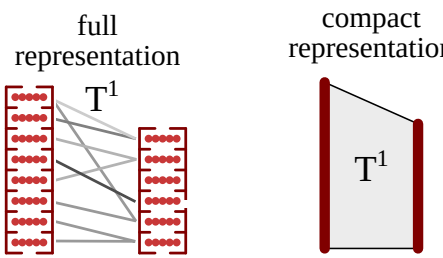
Table of Contents

Optimal Tensor Transport

Based on a published paper at the Association for the Advancement of Artificial Intelligence (AAAI) [Kerdoncuff et al., 2022].



Motivation of Optimal Tensor Transport (OTT)

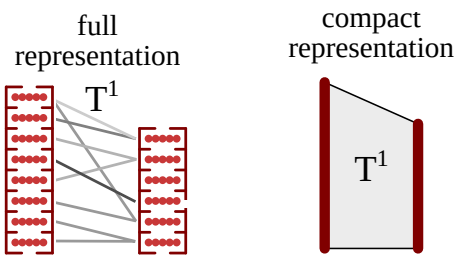


a) **OTT$_1$** (OT) *(F=5 features)*

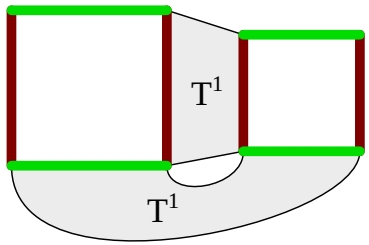
Optimal Transport

$$\mathrm{OT} = \min_{T^1 \in \Pi_{a^1 b^1}} \sum_{i_1=1}^{I_1} \sum_{k_1=1}^{K_1} \mathcal{L}(X_{i_1}, Y_{k_1})\, T^1_{i_1 k_1} \tag{8}$$

Motivation of Optimal Tensor Transport (OTT)



a) **OTT$_1$** (OT) *(F=5 features)*

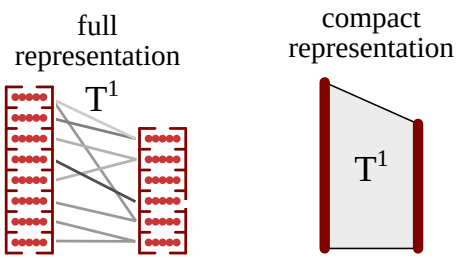


b) **OTT$_{11}$** (GW)

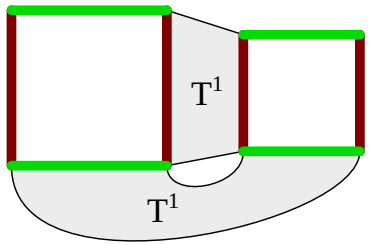
Gromov Wasserstein

$$\mathrm{GW} = \min_{T^1 \in \Pi_{a^1 b^1}} \sum_{i_1, i_2 = 1}^{I_1, I_2} \sum_{k_1, k_2 = 1}^{K_1, K_2} \mathcal{L}(X_{i_1 i_2}, Y_{k_1 k_2})\, T^1_{i_1 k_1}\, T^1_{i_2 k_2} \tag{8}$$

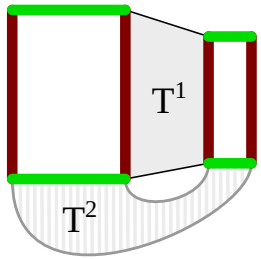
Motivation of Optimal Tensor Transport (OTT)



a) **OTT$_1$** (OT) *(F=5 features)*



b) **OTT$_{11}$** (GW)

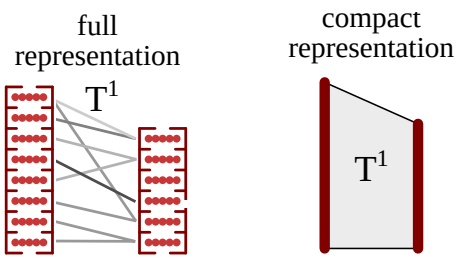


c) **OTT$_{12}$** (Co-OT)

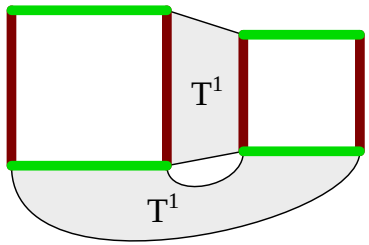
Co-Optimal Transport [Redko et al., 2020]

$$\text{Co-OT} = \min_{T^1 \in \Pi_{a^1 b^1}\, T^2 \in \Pi_{a^2 b^2}} \sum_{i_1, i_2 = 1}^{I_1, I_2} \sum_{k_1, k_2 = 1}^{K_1, K_2} \mathcal{L}(X_{i_1 i_2}, Y_{k_1 k_2})\, T^1_{i_1 k_1}\, T^2_{i_2 k_2} \qquad (8)$$

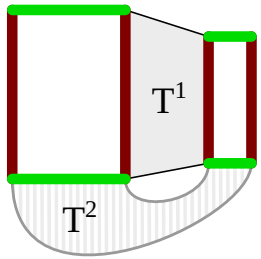
Motivation of Optimal Tensor Transport (OTT)



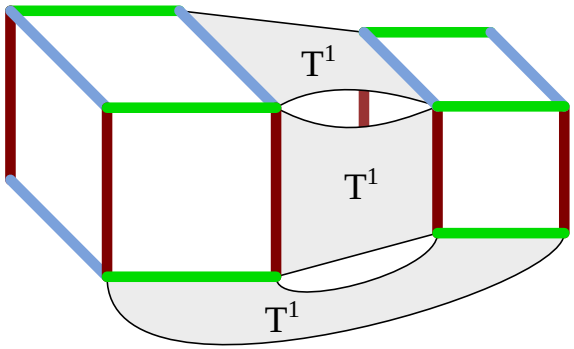
a) **OTT$_1$** (OT) *(F=5 features)*



b) **OTT$_{11}$** (GW)



c) **OTT$_{12}$** (Co-OT)

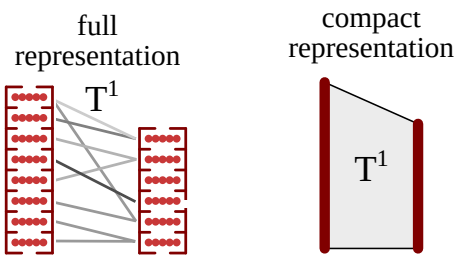


d) **OTT$_{111}$** (triplets)

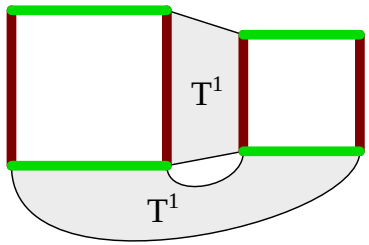
Optimal Tensor Transport with $(1, 1, 1)$

$$\min_{T^1 \in \Pi_{a^1 b^1}} \sum_{i_1,i_2,i_3=1}^{I_1,I_1,I_1} \sum_{k_1,k_2,k_3=1}^{K_1,K_1,K_1} \mathcal{L}(X_{i_1 i_2 i_3}, Y_{k_1 k_2 k_3})\, T^1_{i_1 k_1}\, T^1_{i_2 k_2}\, T^1_{i_3 k_3}$$

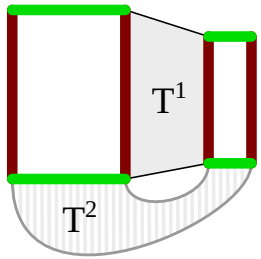
Motivation of Optimal Tensor Transport (OTT)



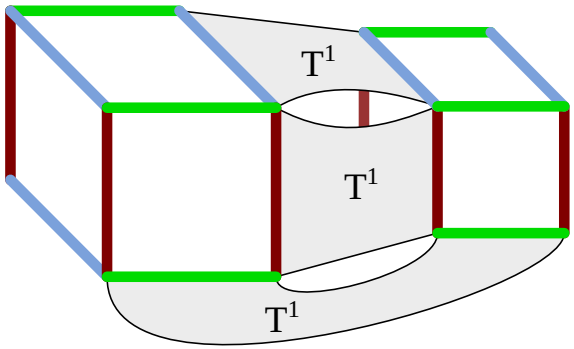
a) **OTT$_1$** (OT) *(F=5 features)*



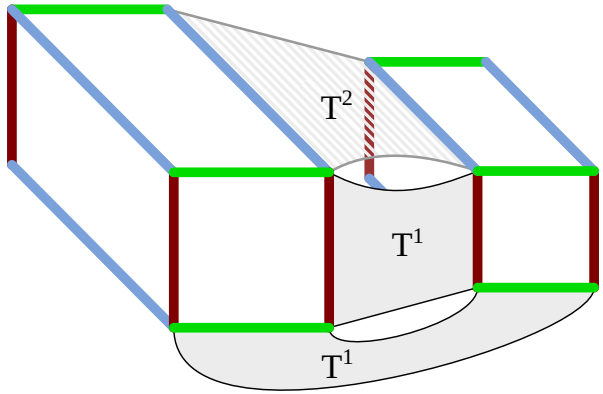
b) **OTT$_{11}$** (GW)



c) **OTT$_{12}$** (Co-OT)



d) **OTT$_{111}$** (triplets)

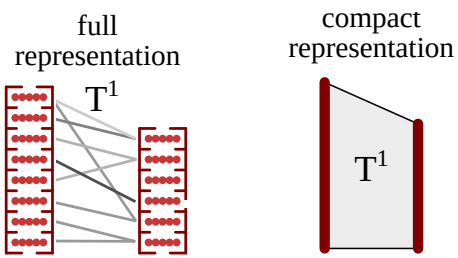


e) **OTT$_{112}$** (GW collections)

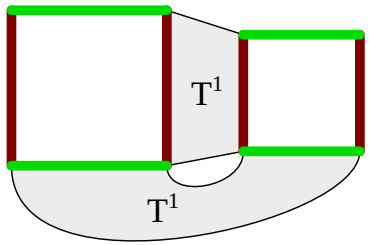
Optimal Tensor Transport with $(1, 1, 2)$

$$\min_{T^1 \in \Pi_{a^1 b^1}\, T^2 \in \Pi_{a^2 b^2}} \sum_{i_1,i_2,i_3=1}^{I_1,I_1,I_2} \sum_{k_1,k_2,k_3=1}^{K_1,K_1,K_2} \mathcal{L}(X_{i_1 i_2 i_3}, Y_{k_1 k_2 k_3})\, T^1_{i_1 k_1}\, T^1_{i_2 k_2}\, T^2_{i_3 k_3}$$

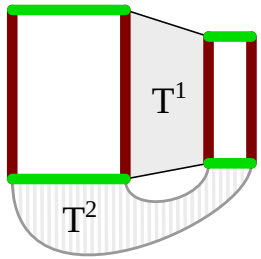
Motivation of Optimal Tensor Transport (OTT)



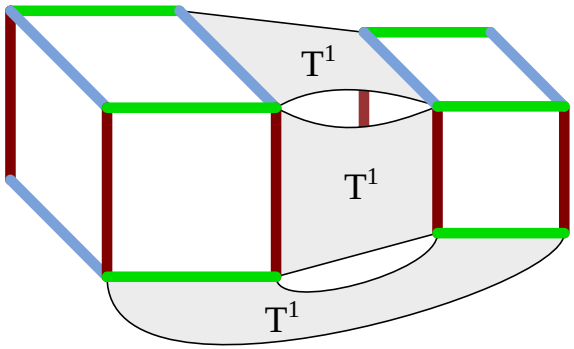
a) **OTT$_1$** (OT) *(F=5 features)*



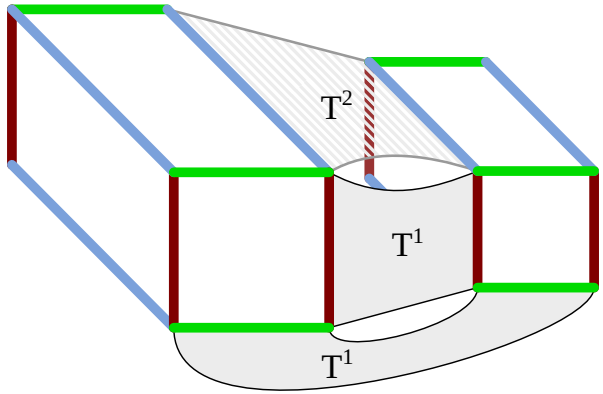
b) **OTT$_{11}$** (GW)



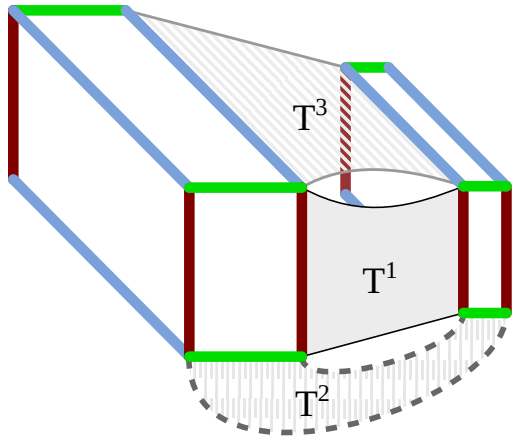
c) **OTT$_{12}$** (Co-OT)



d) **OTT$_{111}$** (triplets)



e) **OTT$_{112}$** (GW collections)



f) **OTT$_{123}$** (triCo-OT)

Optimal Tensor Transport with $(1, 2, 3)$

$$\min_{\forall e\ T^e \in \Pi_{a^e b^e}} \sum_{i_1,i_2,i_3=1}^{I_1,I_2,I_3} \sum_{k_1,k_2,k_3=1}^{K_1,K_2,K_3} \mathcal{L}(X_{i_1 i_2 i_3}, Y_{k_1 k_2 k_3})\, T^1_{i_1 k_1}\, T^2_{i_2 k_2}\, T^3_{i_3 k_3}$$

Stochastic Mirror Descent to solve the OTT problem

Algorithm 8 Alternated Stochastic Mirror Descent algorithm

Require: $(a^e)_{e\in[\![1,E]\!]}$, $(b^e)_{e\in[\![1,E]\!]}$, X, Y, $\mathcal{L}$, M, ϵ

$\forall e \in [\![1, E]\!],\ T^e = a^e b^{e\top}$

for s = 0 **to** S-1 **do**

  for e = 1 **to** E **do**

    $\widehat{\nabla_{T^e}\mathcal{E}_f}$ = M samples of the gradient

    $T^e = \min_{T'\in\Pi(a^e,b^e)} \left\langle \widehat{\nabla_{T^e}\mathcal{E}_f}, T' \right\rangle + \epsilon KL(T'||T^e)$

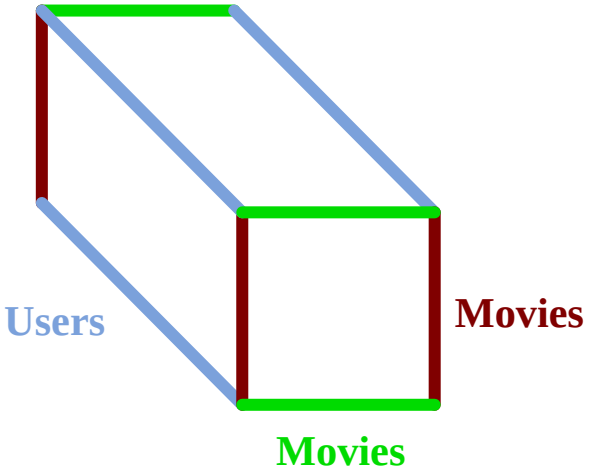
  end for

end for

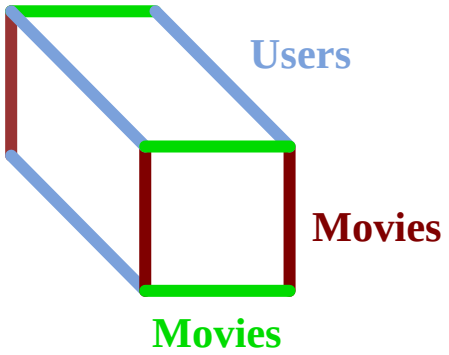
Application to Domain Adaptation

Domain Adaptation: predict the type of movies

Movielens dataset [Harper and Konstan, 2015]



Old movies



New movies

	SVM	S-GWL	GW	Co-OT	OTT
AVG	58.6	63.8	64.9	68.2	**77.3**±3.1

Table of Contents

Conclusion

Contributions in 3 domains

- Ground metric

1. MLOT
   - ✓ Interest of **learning a ground metric** for a DA task.

Conclusion

Contributions in 3 domains

- Ground metric

1. MLOT
   - ✓ Interest of **learning a ground metric** for a DA task.
2. Minimax
   - ✓ General **cutting set** method for minimax OT problems
   - ✓ **Wasserstein stability** definition

Conclusion

Contributions in 3 domains

- Ground metric
- Computation speed

1. MLOT
   - ✓ Interest of **learning a ground metric** for a DA task.
2. Minimax
   - ✓ General **cutting set** method for minimax OT problems
   - ✓ **Wasserstein stability** definition
3. Sampled GW
   - ✓ Fast approximation of the GW distance with **any loss**
   - ✓ Theoretical **convergence bound** to stationary points
   - ✓ **Very fast approximation** of the GW distance using the 1D OT solver

Conclusion

Contributions in 3 domains

- Ground metric
- Computation speed
- General OT framework

1. MLOT
   - ✓ Interest of **learning a ground metric** for a DA task.
2. Minimax
   - ✓ General **cutting set** method for minimax OT problems
   - ✓ **Wasserstein stability** definition
3. Sampled GW
   - ✓ Fast approximation of the GW distance with **any loss**
   - ✓ Theoretical **convergence bound** to stationary points
   - ✓ **Very fast approximation** of the GW distance using the 1D OT solver
4. OTT
   - ✓ New OT formulation to handle **tensors** of arbitrary orders
   - ✓ Adaptation of the previous solver to solve it in a reasonable time

Further works related to the contributions

1. Deep version of MLOT

Further works related to the contributions

1. Deep version of MLOT
2. Extend the use of the Wasserstein stability with a min minimax formulation

Further works related to the contributions

1. Deep version of MLOT
2. Extend the use of the Wasserstein stability with a min minimax formulation
3. Convergence bound of SaGroW^{KL} to stationary points

Further works related to the contributions

1. Deep version of MLOT
2. Extend the use of the Wasserstein stability with a min minimax formulation
3. Convergence bound of SaGroW^{KL} to stationary points
4. Fused-OTT:

Further works related to the contributions

1. Deep version of MLOT
2. Extend the use of the Wasserstein stability with a min minimax formulation
3. Convergence bound of SaGroW^{KL} to stationary points
4. Fused-OTT:

Fused-Gromov Wasserstein:

$$\min_{T \in \Pi(\mu,\nu)} \alpha \sum_{i,k=1}^{I,K} C_{ik}\, T_{ik} + (1-\alpha) \sum_{i,j,k,l=1}^{I,I,K,K} L_{ijkl}\, T_{ik}\, T_{jl} \tag{9}$$

Further works related to the contributions

1. Deep version of MLOT
2. Extend the use of the Wasserstein stability with a min minimax formulation
3. Convergence bound of SaGroW^{KL} to stationary points
4. Fused-OTT:

$$\min_{T \in \Pi(\mu,\nu)} \alpha_1 \sum_{i_1,k_1=1}^{I,K} L^1_{i_1 k_1} T_{i_1 k_1} + \alpha_2 \sum_{i_1,i_2,k_1,k_2=1}^{I,I,K,K} L^2_{i_1 i_2 k_1 k_2} T_{i_1 k_1} T_{i_2 k_2} \tag{8}$$

$$+\alpha_3 \sum_{i_1,i_2,i_3,k_1,k_2,k_3=1}^{I,I,I,K,K,K} L^3_{i_1 i_2 i_3 k_1 k_2 k_3} T_{i_1 k_1} T_{i_2 k_2} T_{i_3 k_3}$$

Fused-Gromov Wasserstein:

$$\min_{T \in \Pi(\mu,\nu)} \alpha \sum_{i,k=1}^{I,K} C_{ik} T_{ik} + (1-\alpha) \sum_{i,j,k,l=1}^{I,I,K,K} L_{ijkl} T_{ik} T_{jl} \tag{9}$$

Some perspectives for the OT's application to ML

1. Continuous GW
   - ✓ Continuous version using PoGroW already done
     - SaGroW with an approximation for the continuous OT problem

Some perspectives for the OT's application to ML

1. Continuous GW
   - ✓ Continuous version using PoGroW already done
     - SaGroW with an approximation for the continuous OT problem
2. GW $\Leftrightarrow$ Linear WGAN
   - Fused WGAN

Some perspectives for the OT's application to ML

1. Continuous GW
   - ✓ Continuous version using PoGroW already done
     - SaGroW with an approximation for the continuous OT problem
2. GW $\Leftrightarrow$ Linear WGAN
   - Fused WGAN
3. Sparse solution of the GW problem
   - ✓ Concave case already known
     - Under which condition on $\mathcal{L}$?

Some perspectives for the OT's application to ML

1. Continuous GW
   - ✓ Continuous version using PoGroW already done
     - SaGroW with an approximation for the continuous OT problem
2. GW $\Leftrightarrow$ Linear WGAN
   - Fused WGAN
3. Sparse solution of the GW problem
   - ✓ Concave case already known
     - Under which condition on $\mathcal{L}$?
4. Relation of OT formulation to visualization and clustering methods
   - ✓ Strong link between PCA and Wasserstein barycenter
   - ✓ UMAP/t-STE and Gromov barycenter
   - ✓ Kmeans and Wasserstein barycenter
     - TAUDoS project: Fused-Kmeans for the distillation of Recurrent Neural Networks

Thank you for your attention

Published papers

- KERDONCUFF Tanguy, EMONET Rémi and SEBBAN Marc
  Metric Learning in Optimal Transport for Domain Adaptation
  IJCAI 2020
- DHOUIB Sofien, REDKO Ievgen, KERDONCUFF Tanguy, EMONET, Rémi, and SEBBAN Marc
  A Swiss Army Knife for Minimax Optimal Transport
  ICML 2021
- KERDONCUFF Tanguy, EMONET Rémi, and SEBBAN Marc
  Sampled Gromov Wasserstein
  MLJ 2021
- KERDONCUFF Tanguy, PERROT Michael, EMONET Rémi, and SEBBAN Marc
  Optimal Tensor Transport
  AAAI 2022

Biblio I

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. (2017). Optimal transport for domain adaptation. *PAMI*.

Cuturi, M. (2013). Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*.

Dhouib, S., Redko, I., Kerdoncuff, T., Emonet, R., and Sebban, M. (2020). A swiss army knife for minimax optimal transport. In *International Conference on Machine Learning*.

Flamary, R. and Courty, N. (2017). Pot python optimal transport library.

Gong, B., Shi, Y., Sha, F., and Grauman, K. (2012). Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*.

Kantorovich, L. (1942). On the translocation of masses. *Doklady of the Academy of Sciences of the USSR*, 37:199–201.

Kerdoncuff, Tanguy Michaëel, P., Emonet, R., and Sebban, M. (2022). Optimal tensor transport. In *AAAI*.

Biblio II

Kerdoncuff, T., Emonet, R., and Sebban, M. (2020). Metric learning in optimal transport for domain adaptation. In Bessiere, C., editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 2162–2168. International Joint Conferences on Artificial Intelligence Organization. Main track.

Kerdoncuff, T., Emonet, R., and Sebban, M. (2021). Sampled gromov wasserstein. *Machine Learning*.

Memoli, F. (2007). On the use of Gromov-Hausdorff Distances for Shape Comparison. In Botsch, M., Pajarola, R., Chen, B., and Zwicker, M., editors, *Eurographics Symposium on Point-Based Graphics*. The Eurographics Association.

Monge, G. (1781). Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Académie royale des sciences de Paris*.

Mutapcic, A. and Boyd, S. P. (2009). Cutting-set methods for robust convex optimization with pessimizing oracles. *Optimization Methods and Software*.

Peyré, G., Cuturi, M., and Solomon, J. (2016). Gromov-wasserstein averaging of kernel and distance matrices. In *International Conference on Machine Learning*.

Rangarajan, A., Yuille, A., and Mjolsness, E. (1999). Convergence properties of the softassign quadratic assignment algorithm. *Neural Computation*.

Biblio III

Reddi, S. J., Sra, S., Póczos, B., and Smola, A. (2016). Stochastic frank-wolfe methods for nonconvex optimization. In *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*.

Redko, I., Vayer, T., Flamary, R., and Courty, N. (2020). Co-optimal transport. In *NeurIPS 2020-Thirty-four Conference on Neural Information Processing Systems*.

Sinkhorn, R. and Knopp, P. (1967). Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*.

Tan, L. H., Chan, A. H., Kay, P., Khong, P.-L., Yip, L. K., and Luke, K.-K. (2008). Language affects patterns of brain activation associated with perceptual decision. *Proceedings of the National Academy of Sciences*, 105(10):4004–4009.

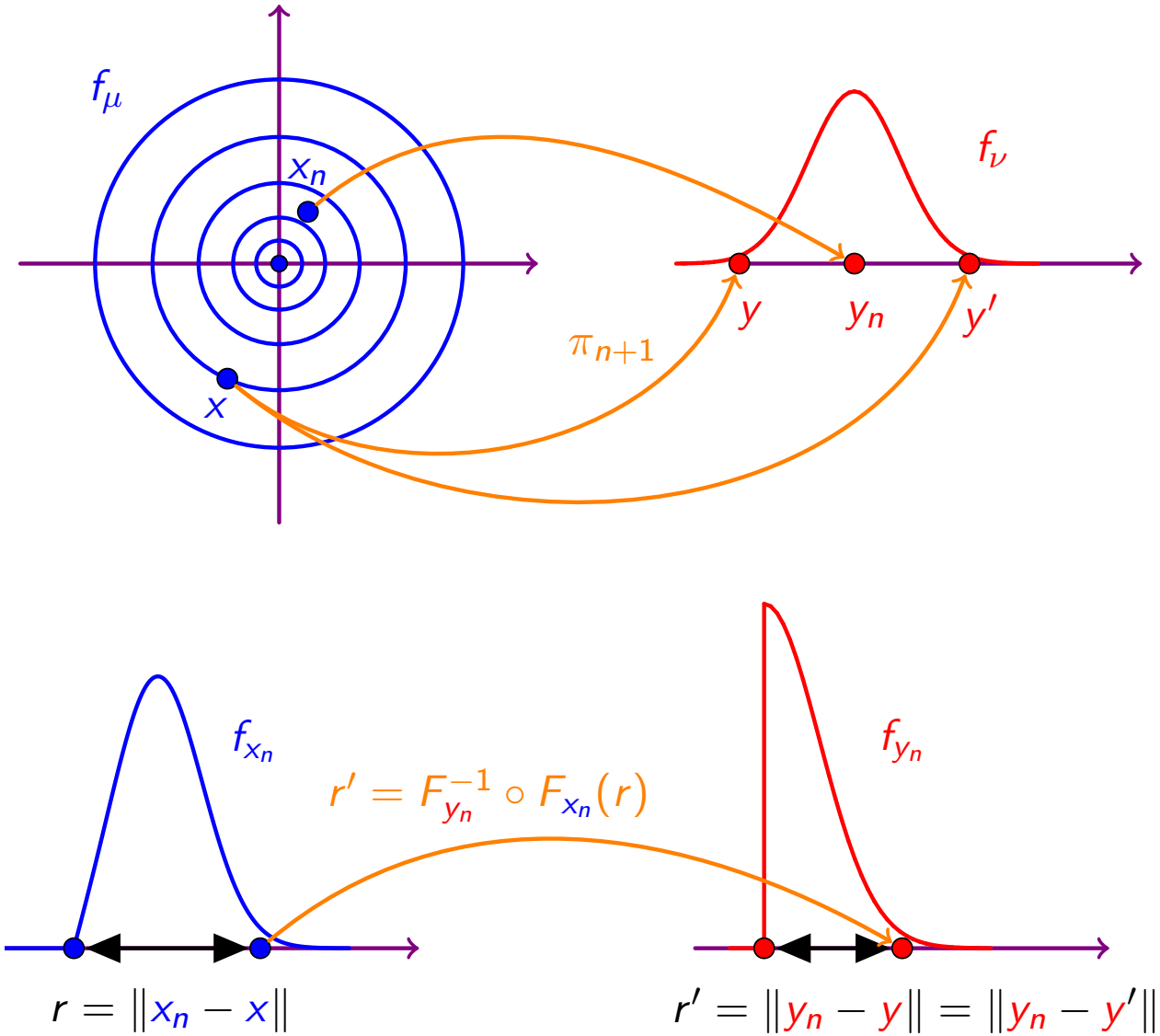
Vayer, T., Flamary, R., Tavenard, R., Chapel, L., and Courty, N. (2019). Sliced gromov-wasserstein. In *NeurIPS 2019-Thirty-third Conference on Neural Information Processing Systems*.

Villani, C. (2008). *Optimal transport: old and new*.

Weinberger, K. Q. and Saul, L. K. (2009). Distance metric learning for large margin nearest neighbor classification. *JMLR*.

Xu, H., Luo, D., and Carin, L. (2019). Scalable gromov-wasserstein learning for graph partitioning and matching. In *Advances in neural information processing systems*.

Continuous GW



Bound for convergence: concave case

- The problem is concave notably when T when $\mathcal{L}(x, y) = (x - y)^2$, $\mathcal{C}^{\mathcal{X}}$ and $\mathcal{C}^{\mathcal{Y}}$ are the euclidean distance.

Theorem (Concave case)

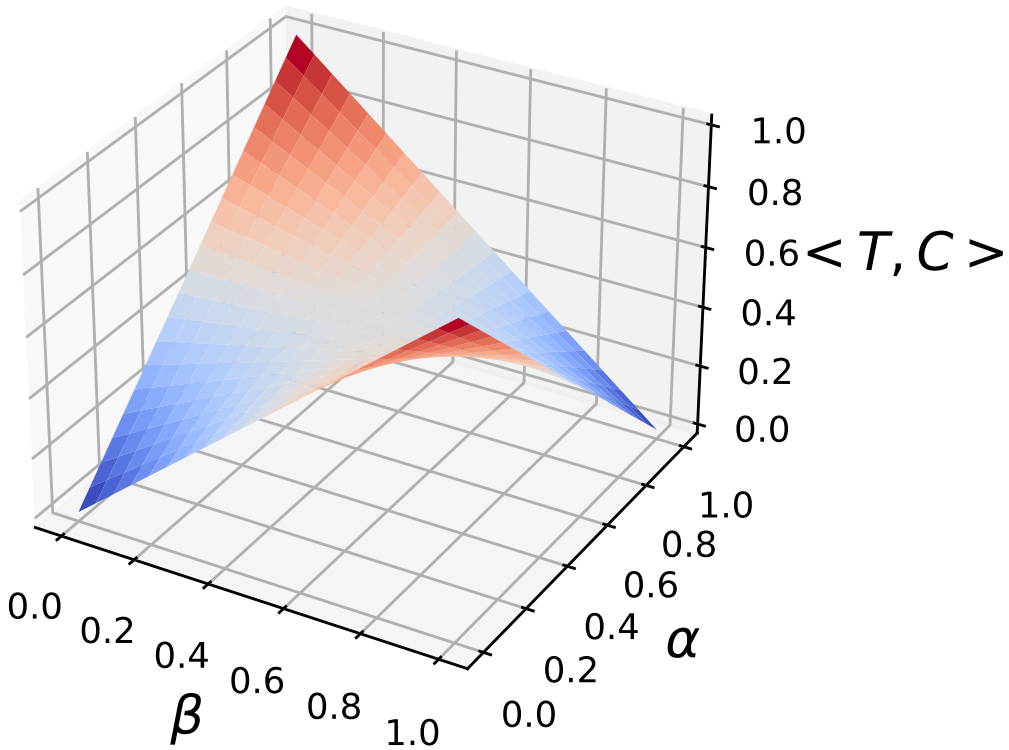
With the same notations as in the previous Theorem with the entropy regularization parameter ϵ_s that may now change along the iterations s, when L yields a concave GW problem, the following bound holds:

$$\mathbb{E}\left(G\left(\overline{T}\right)\right) \leq \frac{\mathcal{E}(T_0) - \mathcal{E}(T^*)}{2S} + B\sqrt{\frac{2N}{M}} + \frac{1}{S}\sum_{s=0}^{S-1} \epsilon_s log(N).$$

Intuition of the OT Minimax problem

The Minimax problem

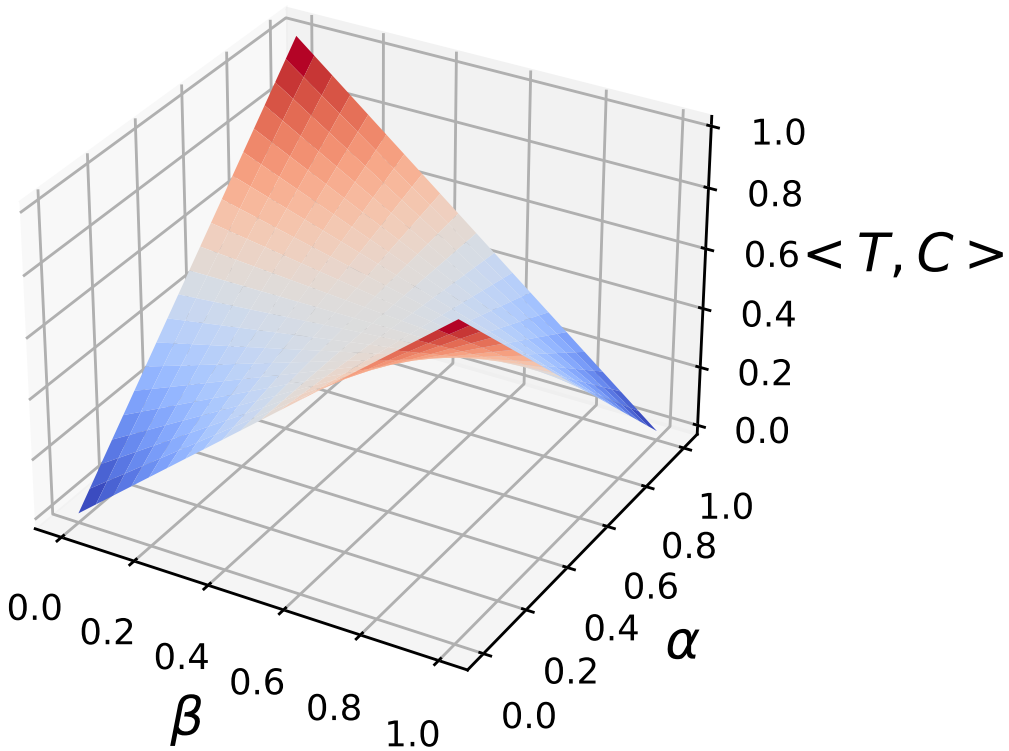
$$\mathcal{W}_{\mathcal{C}} = \min_{T \in \Pi} \max_{C \in \mathcal{C}} \langle T, C \rangle \quad (10)$$



Intuition of the OT Minimax problem

Cost matrices

- $C_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$
- $C_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
- $\mathcal{C} = \alpha C_1 + (1-\alpha) C_2$



Intuition of the OT Minimax problem

Cost matrices

- $C_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$
- $C_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
- $\mathcal{C} = \alpha C_1 + (1-\alpha) C_2$

Associated Transport Plan

- $T_1^* = \begin{pmatrix} 0 & 0.5 \\ 0.5 & 0 \end{pmatrix}$
- $T_2^* = \begin{pmatrix} 0.5 & 0 \\ 0 & 0.5 \end{pmatrix}$
- $\Pi = \beta T_1^* + (1-\beta) T_2^*$