L'IA au LMAC, Sciences de Données avec Transport Optimal

Mokhtar Z. Alaya

ALLIANCE SORBONNE UNIVERSITÉ

utc Recherche

LMAC

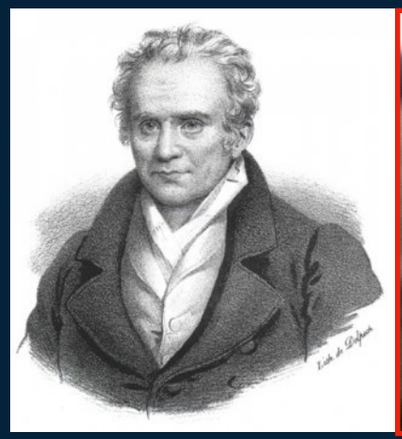
Journée Scientifique, Chaire SAFE AI

UTC, Octobre 2022

1. What is optimal transport (OT)?

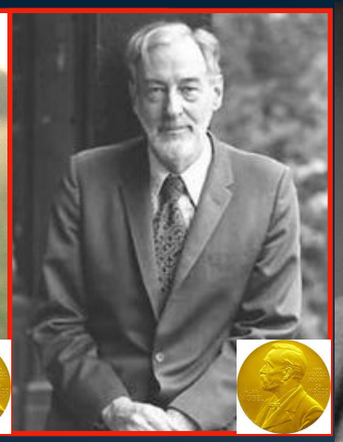
OT is …

A method for comparing probability distributions with the ability to incorporate spatial information.



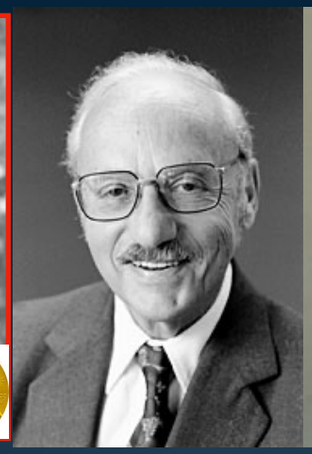
G. Monge
(1746 - 1818)

L. Kantorovich
(1912 - 1986)

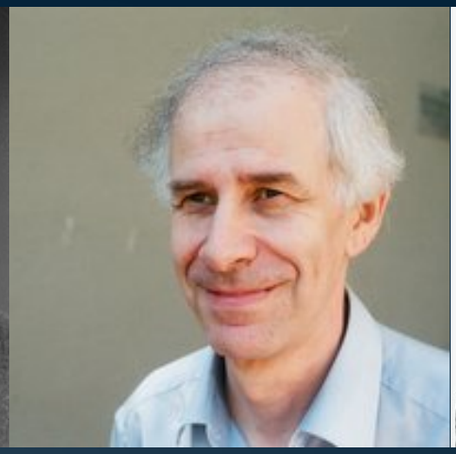


T. Koopmans
(1910 - 1985)

Nobel Prize '75



G. Dantzig
(1914 - 2005)



Y. Brenier

F. Otto

R. McCann

C. Villani

Fields'10

A. Figalli

Fields'18

Origin: Monge Problem (1781)

MÉMOIRE
SUR LA
THÉORIE DES DÉBLAIS
ET DES REMBLAIS.

Par M. MONGE.

LORSQU'ON doit transporter des terres d'un lieu dans un autre, on a coutume de donner le nom de *Déblai* au volume des terres que l'on doit transporter, & le nom de *Remblai* à l'espace qu'elles doivent occuper après le transport.

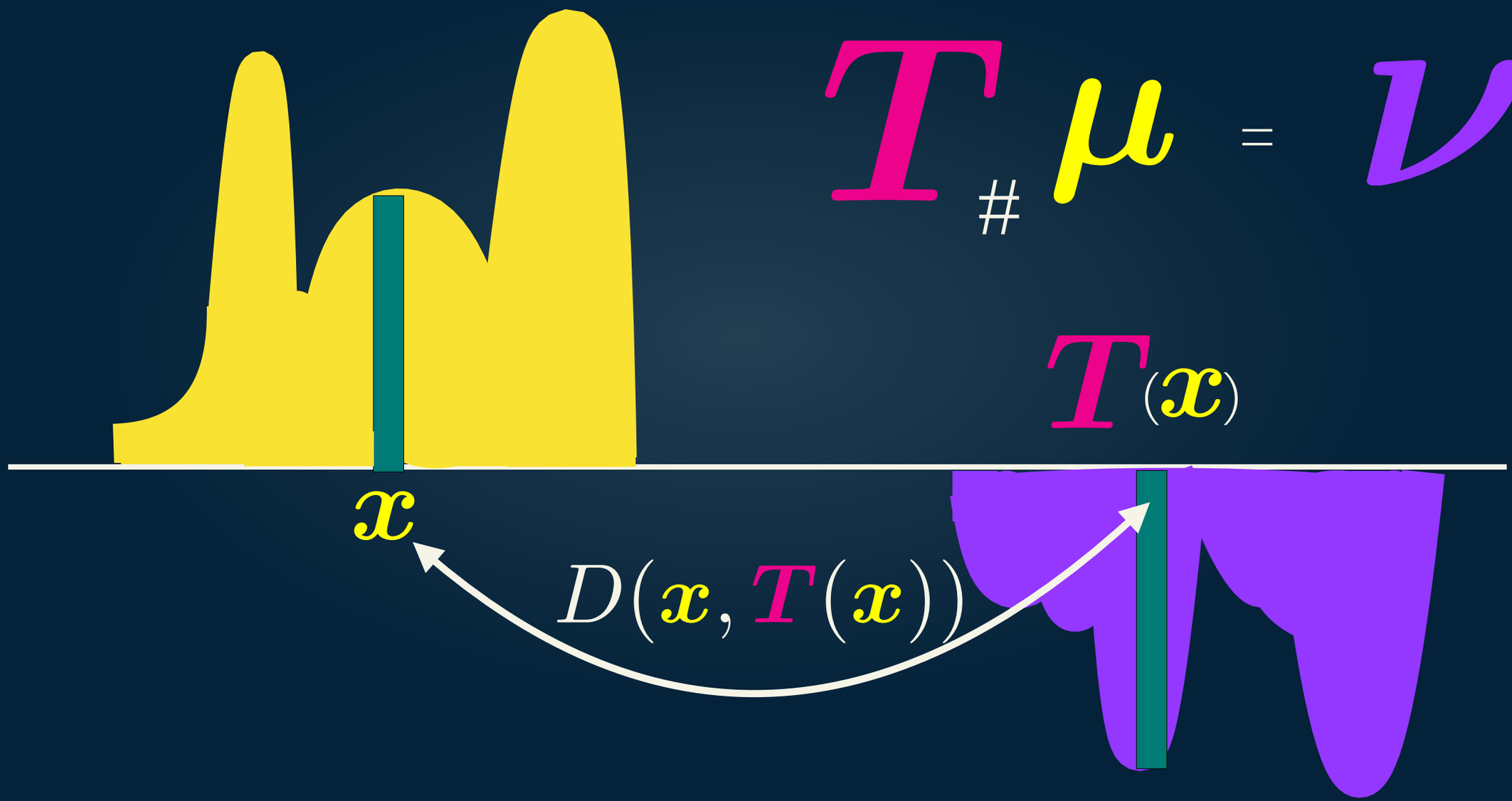
Monge Problem (1781)

- How to move dirt from one place (**déblai**) to another (**remblai**) while minimizing the effort?
- Find a mapping T between the two distributions of mass (**transport**).
- Optimize with respect to a displacement cost (**optimal**).

Monge Problem (1781)

- The mapping T must **push-forward** the **"déblai"** measure towards the **"remblai"**.

$$T_{\#}\mu = \nu$$



Monge Problem (1781)

- Monge formulation aim at finding a mapping $\boldsymbol{T}$ such that:

$$\inf_{\boldsymbol{T}_{\#}\boldsymbol{\mu}=\boldsymbol{\nu}} \int \boldsymbol{C}(\boldsymbol{x}, \boldsymbol{T}(\boldsymbol{x}))\boldsymbol{\mu}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$$

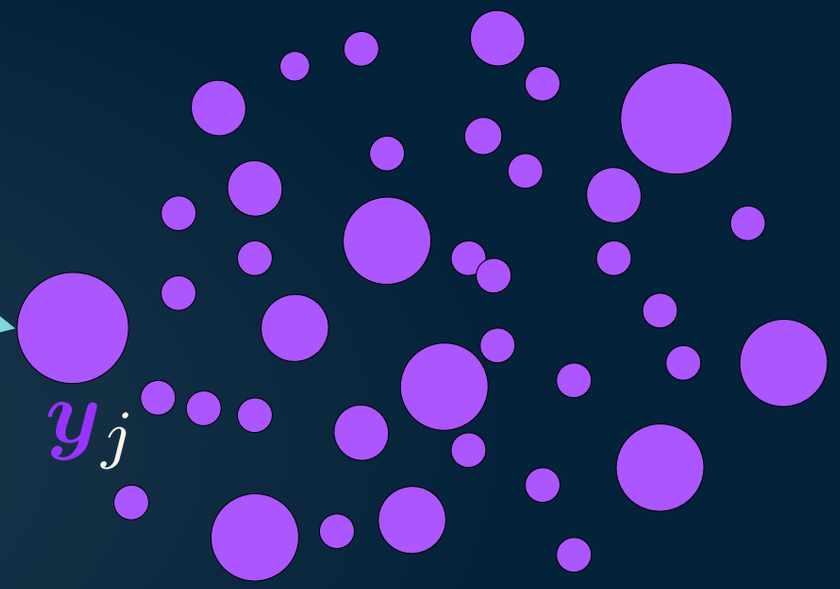
- Mapping $\boldsymbol{T}$ does not exist in the general case.
- Brenier, 1991 proved existence and unicity of the Monge map for Euclidean cost and distributions with densities.

Discrete OT Framework

$$\boldsymbol{\nu} = \sum_{j=1}^{m} \boldsymbol{\nu}_j \delta_{\boldsymbol{y}_j}$$

$\boldsymbol{x}_i$

$D^p(\boldsymbol{x}_i, \boldsymbol{y}_j)$

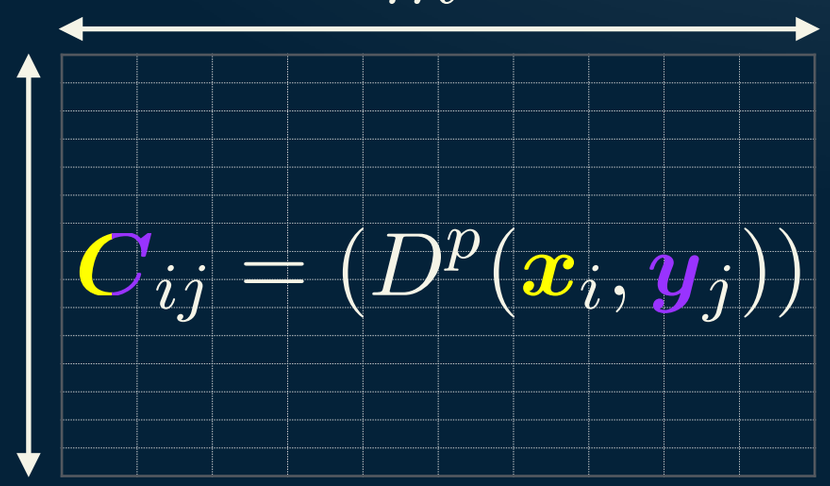


$\boldsymbol{y}_j$

$$\boldsymbol{\mu} = \sum_{i=1}^{n} \boldsymbol{\mu}_i \delta_{\boldsymbol{x}_i}$$

$\boldsymbol{C}$

Cost Matrix

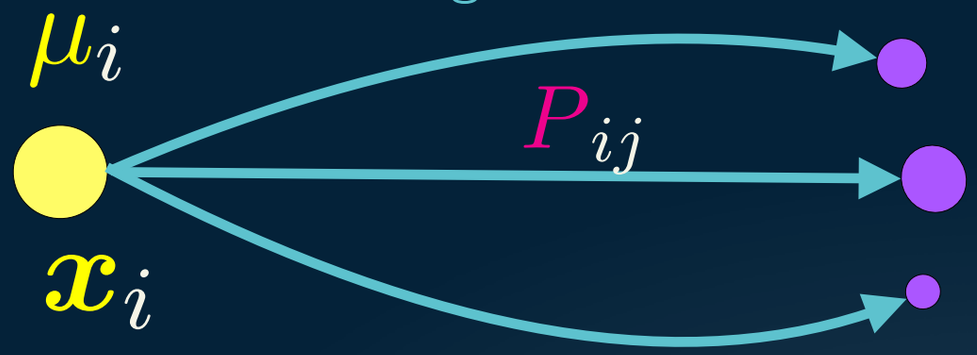


$$\min_{\boldsymbol{T}_{\#}\boldsymbol{\mu}=\boldsymbol{\nu}} \sum_i \boldsymbol{C}(\boldsymbol{x}_i, \boldsymbol{T}(\boldsymbol{x}_i))\boldsymbol{\mu}_i$$

$$\forall j \in \{1, \ldots, m\}, \boldsymbol{\nu}_j = \sum_{i:\boldsymbol{T}(\boldsymbol{x}_i)=\boldsymbol{y}_j} \boldsymbol{\mu}_i$$

Kantorovich's Formula (1942)

Relaxed: Fractional Assignments



$$\mu_i = \sum_{i=1}^{m} P_{ij}$$

- Focus on where the mass goes, allow splitting.
- Applications mainly for resource allocation problems.

$$\min_{\boldsymbol{\gamma} \in \boldsymbol{\Pi}_{\mathrm{con}}(\boldsymbol{\mu}, \boldsymbol{\nu})} \int_{\mathbb{R}^n \times \mathbb{R}^m} \boldsymbol{C}(\boldsymbol{x}, \boldsymbol{y}) \boldsymbol{\gamma}(\boldsymbol{x}, \boldsymbol{y}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y}$$

Probabilistic couplings set (Transport Polytope)

$$\boldsymbol{\Pi}_{\mathrm{con}}(\boldsymbol{\mu}, \boldsymbol{\nu}) = \{\boldsymbol{\gamma} \geq 0, \int_{\mathbb{R}^m} \boldsymbol{\gamma}(\boldsymbol{x}, \boldsymbol{y}) \mathrm{d}\boldsymbol{y} = \boldsymbol{\mu}, \int_{\mathbb{R}^n} \boldsymbol{\gamma}(\boldsymbol{x}, \boldsymbol{y}) \mathrm{d}\boldsymbol{x} = \boldsymbol{\nu}\}$$

Mass conservation constraints

Discrete Version

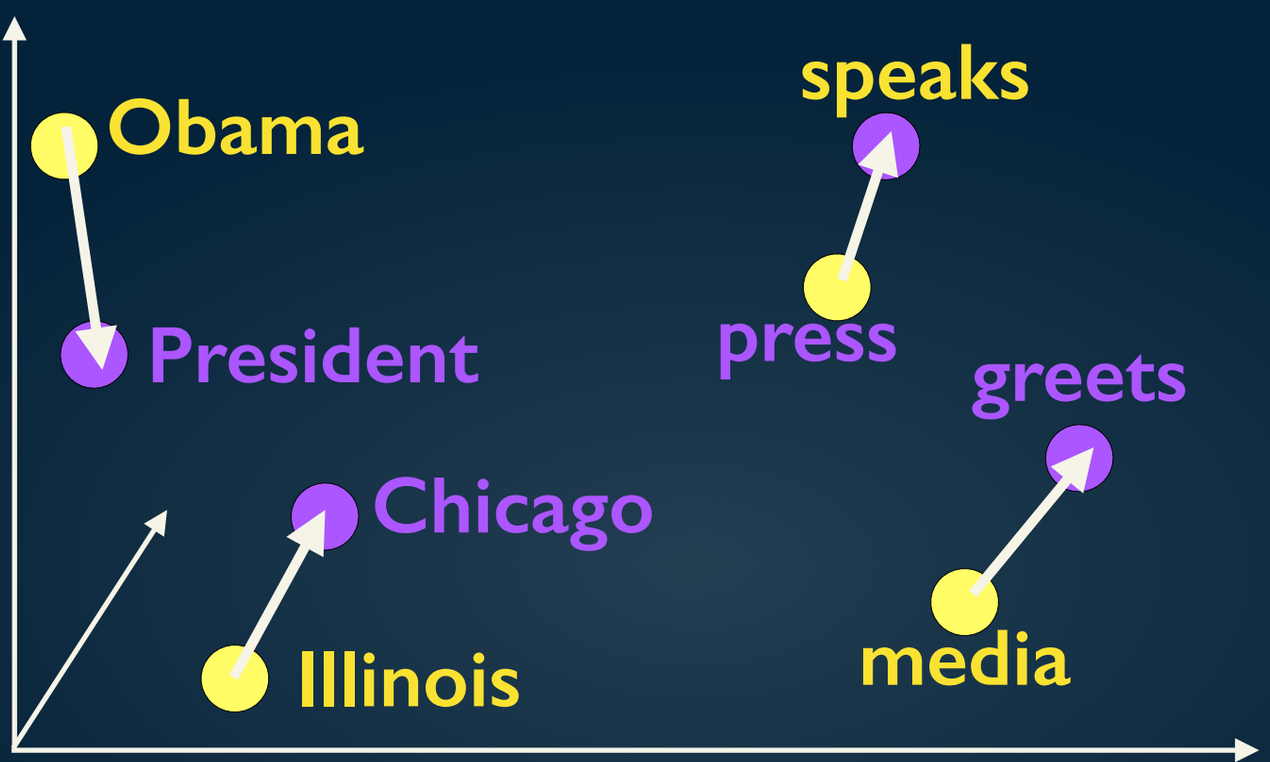
$$\boldsymbol{\Pi}_{\mathrm{dis}}(\boldsymbol{\mu}, \boldsymbol{\nu}) = \{\boldsymbol{P} \in \mathbb{R}_{+}^{n \times m}, \boldsymbol{P} \mathbf{1}_m = \boldsymbol{\mu}, \boldsymbol{P}^{\top} \mathbf{1}_n = \boldsymbol{\nu}\}$$

A first simple examples

Matching words embeddings

Document 1

Obama speaks to the **media** in **Illinois**



Document 2

The **President greets** the **press** In **Chicago**

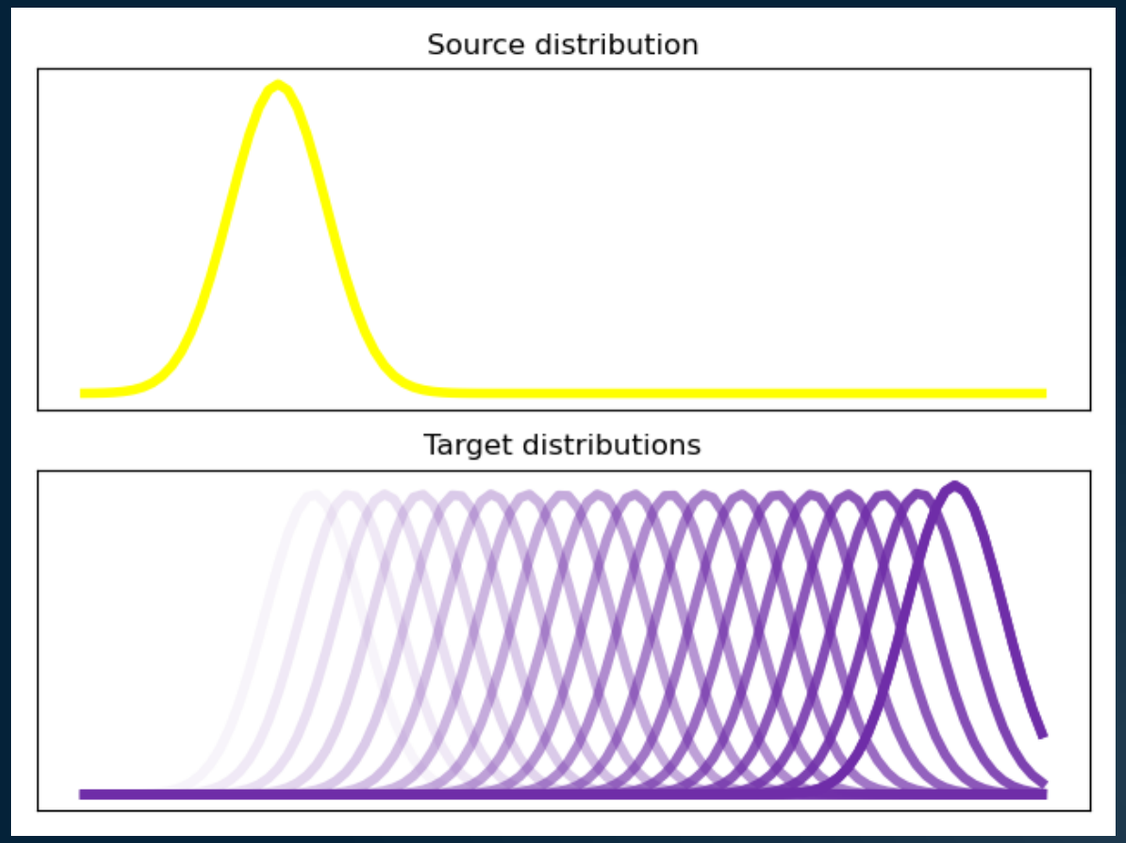
Word Mover's Distance avec Word2vec embeddings
[Kusner et al, 2015 (ICML)]

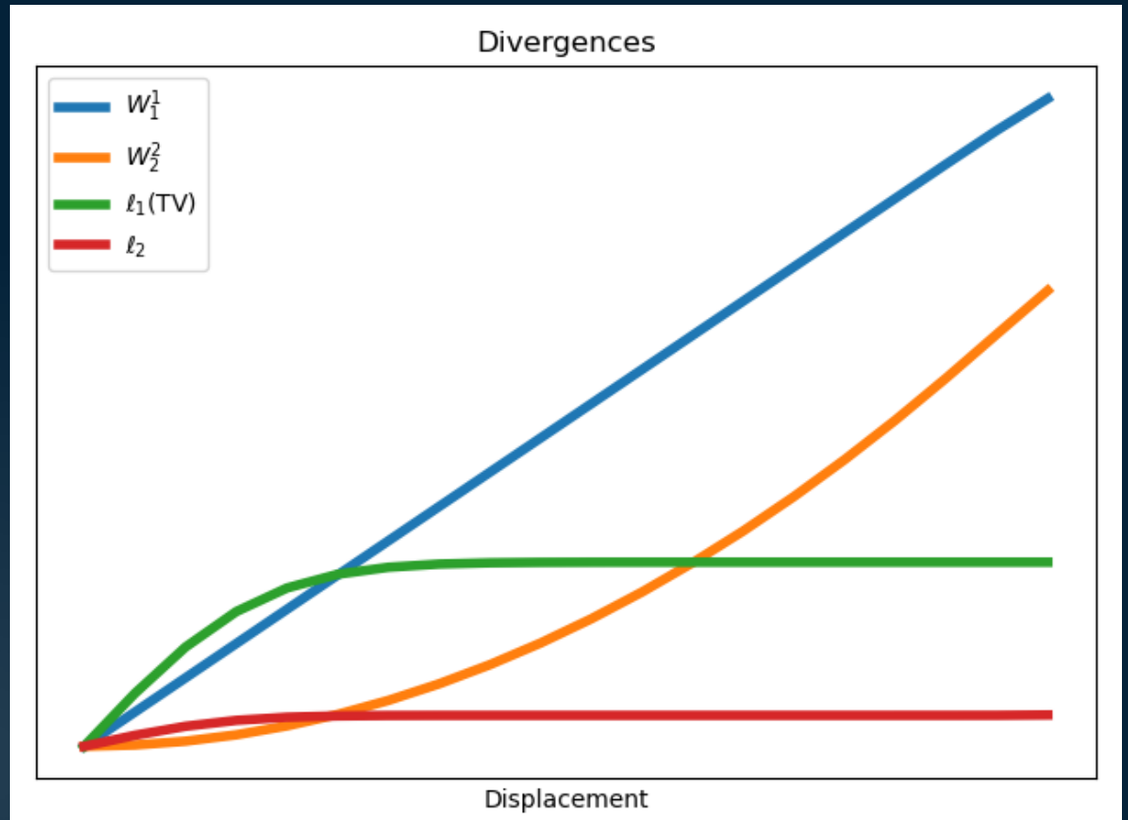
- Words are embedded in a high-dimensional space with deep neural networks.
- Matching two documents in an OT problem, with the Euclidean distance in the embedded space.

Color transfer

Wasserstein distance

Wasserstein distance





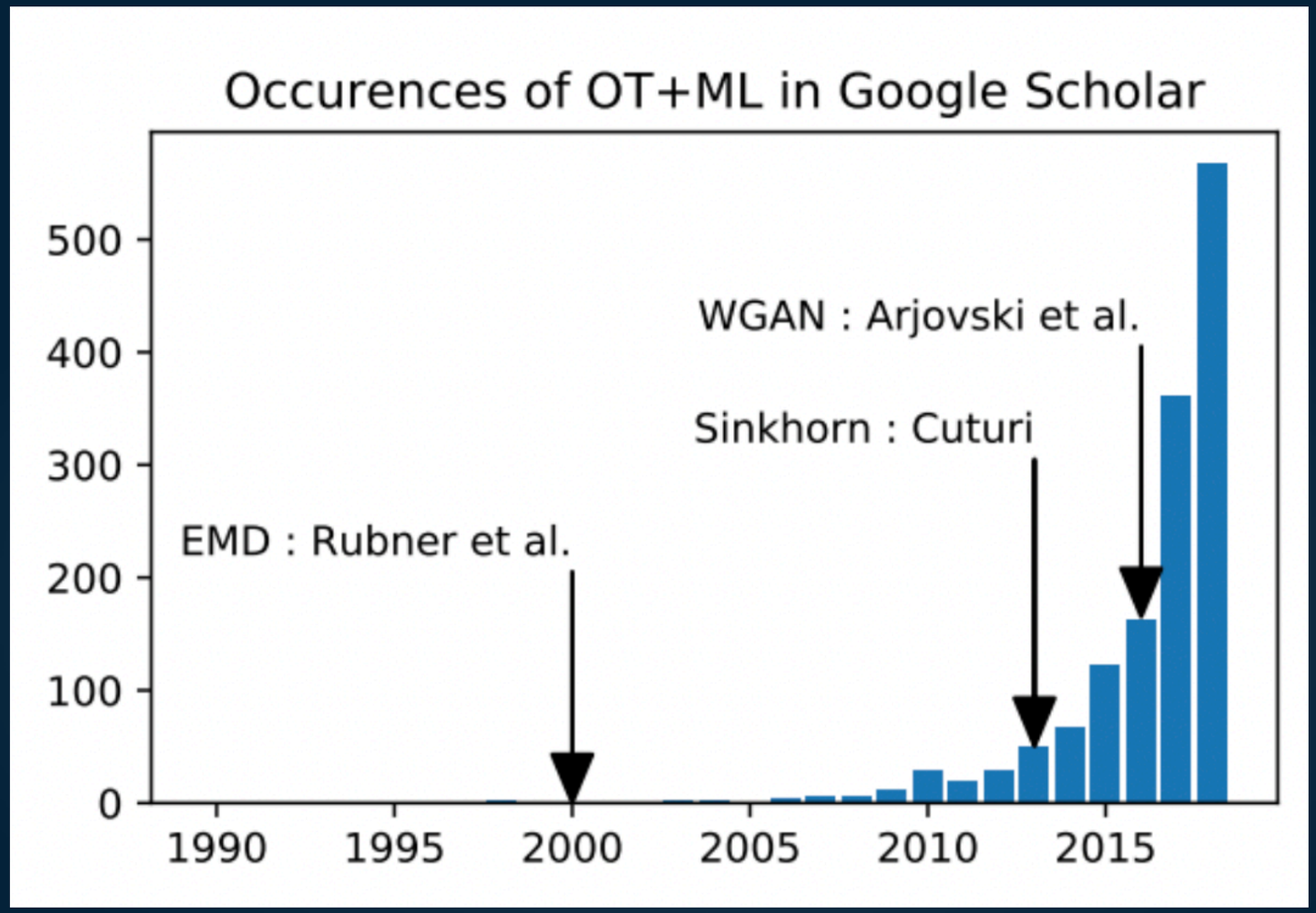
Monge-Kantorovich / Wasserstein Distance

$$\mathcal{W}_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \min_{\boldsymbol{\gamma} \in \boldsymbol{\Pi}_{\text{con}}(\boldsymbol{\mu}, \boldsymbol{\nu})} \int_{\mathbb{R}^n \times \mathbb{R}^m} \boldsymbol{C}(\boldsymbol{x}, \boldsymbol{y}) \boldsymbol{\gamma}(\boldsymbol{x}, \boldsymbol{y}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} = \mathbb{E}_{(\boldsymbol{x}, \boldsymbol{y}) \sim \boldsymbol{\gamma}}[\, \boldsymbol{C}(x, y)]$$

- Do not need the distributions to have overlapping support.
- Works for continuous and discrete distributions (histograms, empirical).

2. How can it be used in data science?

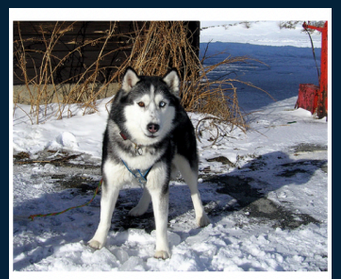
History of OT for machine learning



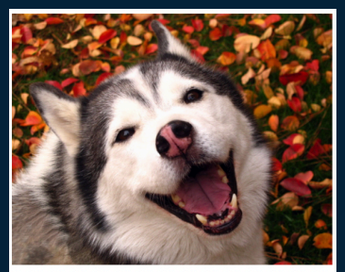
[R. Flamary, 2019 (HDR)]

- Recently introduced to ML (well know in image processing since 2000).
- Computational OT allows numerous applications (regularization).
- Deep learning boost (numerical optimisation and GAN).

Wasserstein distance as a multi-label loss



Siberian husky



Eskimo dog

Flickr user tags: street, parade, dragon
Predictions: people, protest, parade

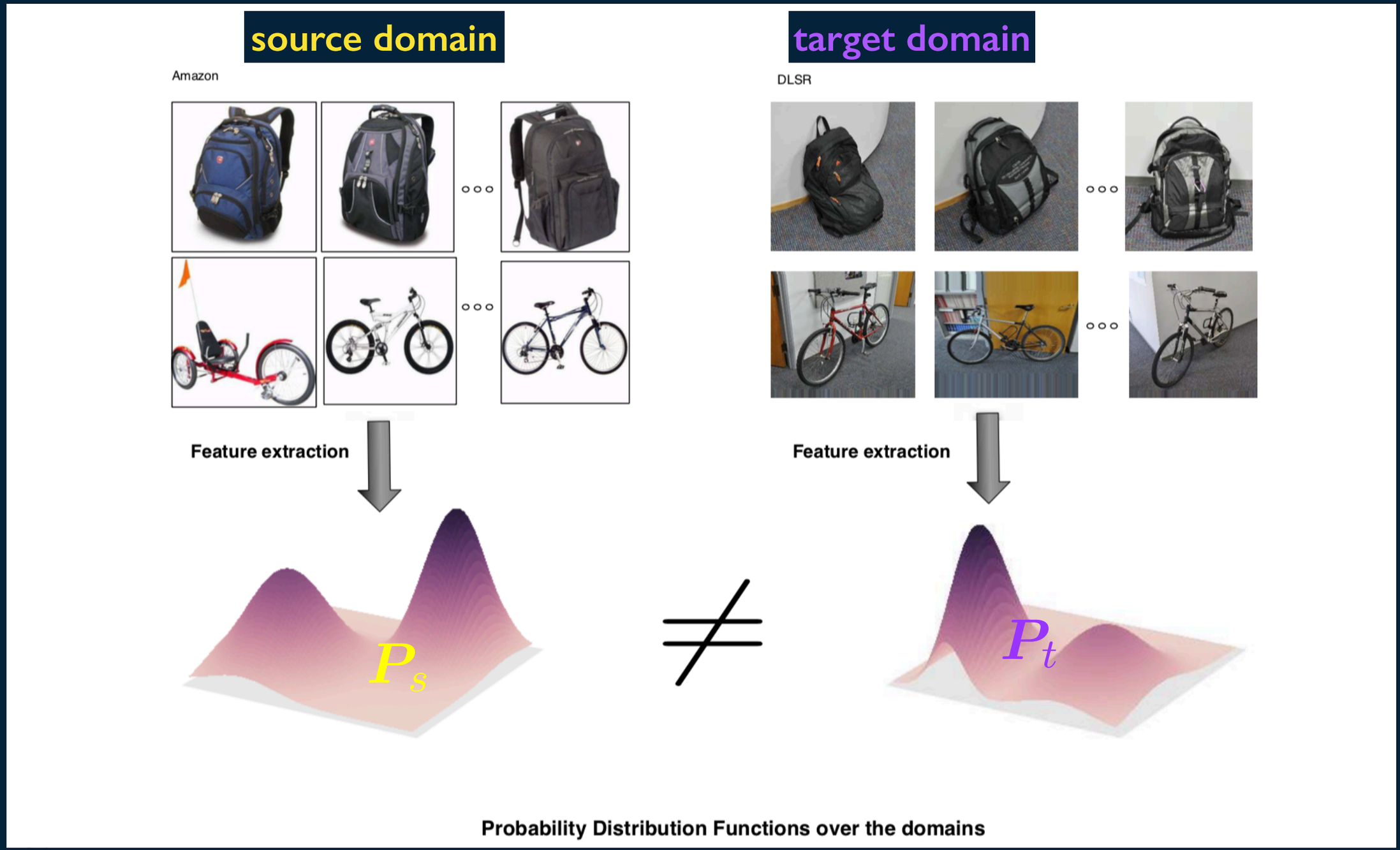
Flickr user tags: water, boat, reflection, sunshine
Predictions: water, river, lake, summer

Leveraging output space structure [Frogner et al., 2015, (NeurIPS)]

- Classes of a multiclass (multi-label) problem have structure.
- Takes into account semantic of classes in the output distribution probability.
- Error in ``similar'' class is less penalized than to dissimilar one .
- Can be represented as a Wasserstein distance between true label and output a model.
- Ground metric represent the distance between classes

$$\min_{f_{\boldsymbol{\theta}}} \frac{1}{n} \sum_{i=1}^{n} W_1^1 \left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i), \boldsymbol{y}_i \right)$$

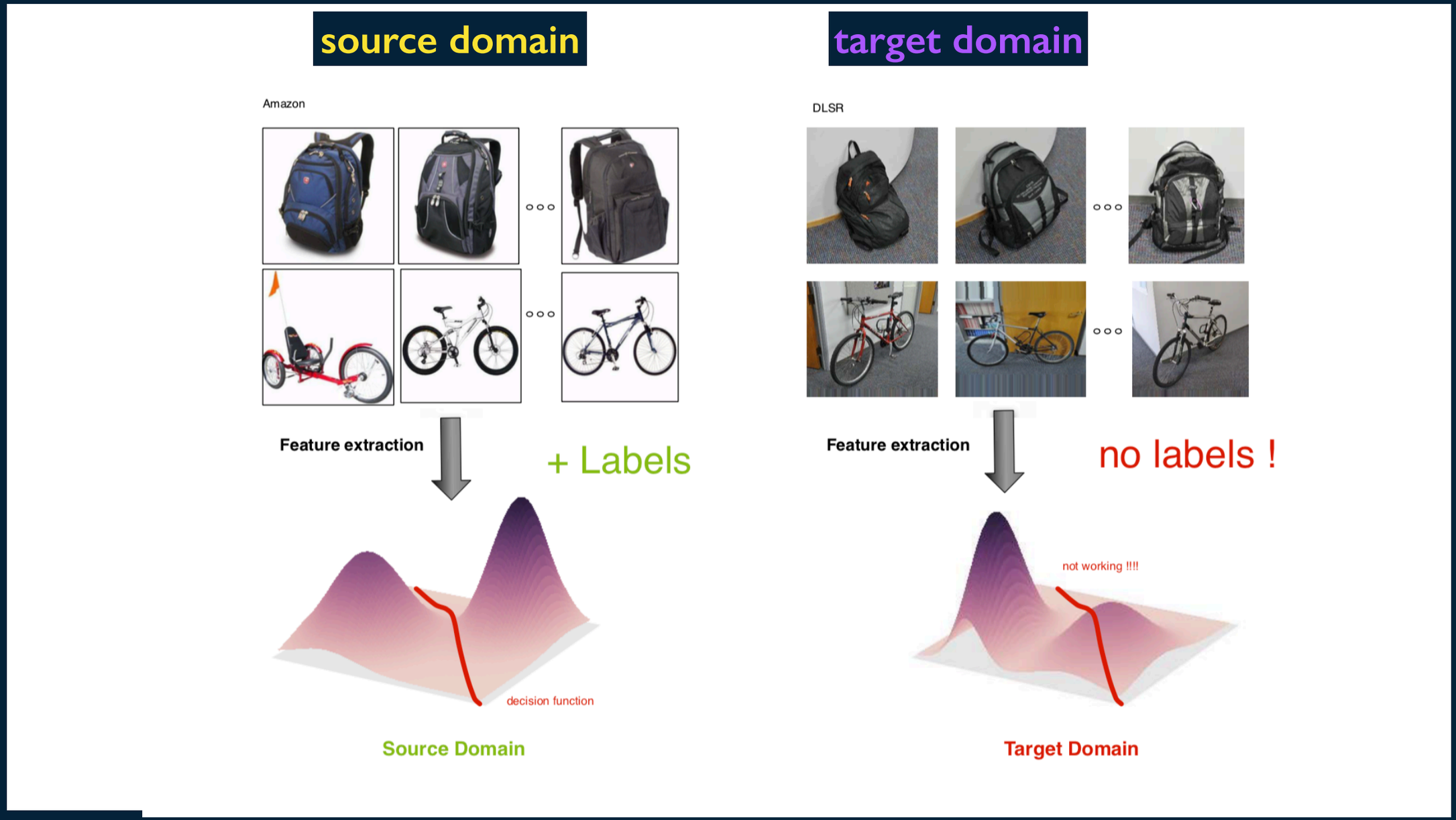
Domain Adaptation Problem



- We Classification problem with data coming from different source (domains).
- Distributions are different but related.

[Credit image: N. Courty]

Domain Adaptation Problem

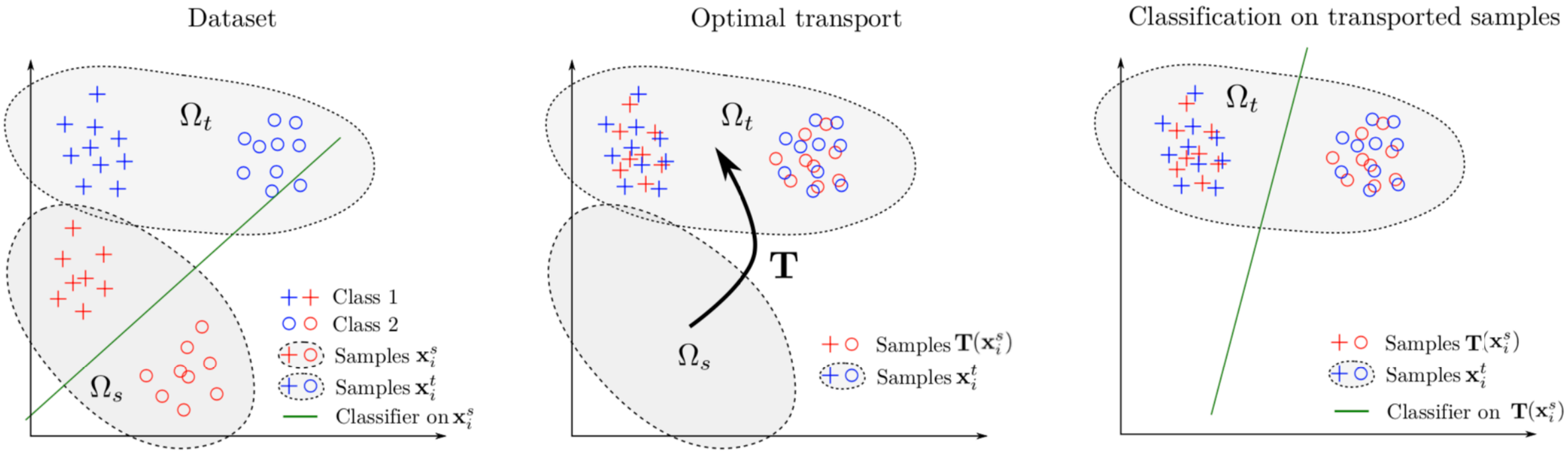


Problems

- Labels only available in the source domain, and classification is conducted in th target domain.
- Classifier on the source domain data performs badly in the target domain.

[Credit image: N. Courty]

OT for Domain Adaptation



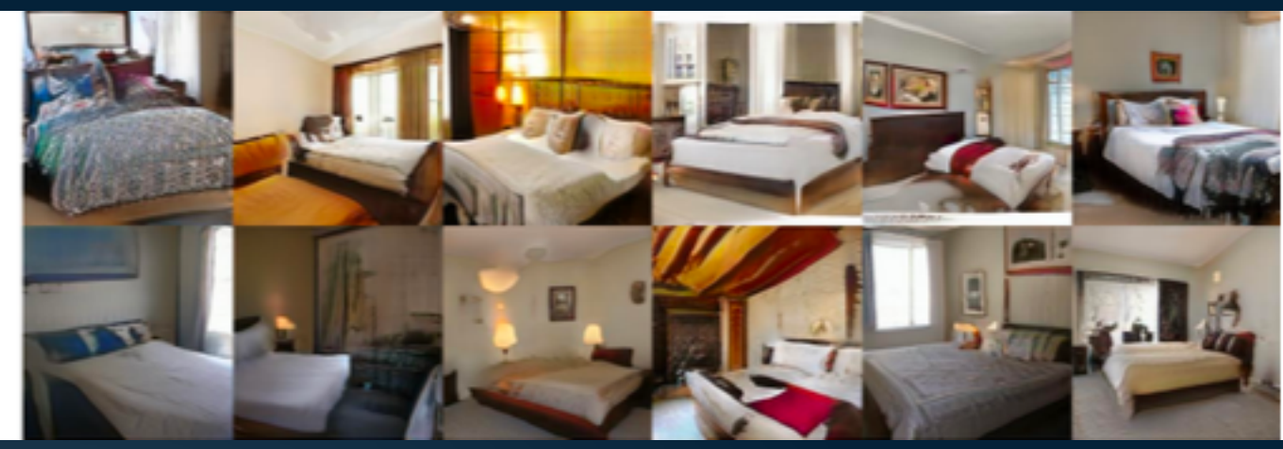
- There exist a transport T in the feature space between the two domains.
- The transport preserves the conditional distributions:

$$P_s[y|x_s] = P_t[y|T(x_s)]$$

3-step strategy [Courty et al., 2017]

1. Estimate optimal transport between distributions.
2. Transport the training samples onto the target distribution using barycentric mapping [Ferradans et al., 2013].
3. Learn a classifier on the transported training samples.

Wasserstein loss for generative modelling



Generative modelling as a matching distribution problem

- Learn a model that maps random vector to target space.
- Distribution of the model is targeted to be similar to the learning samples.
- Similarity as Wasserstein sense [Arjovsky et al. 2017, Deshpande et al. 2018, Nguyen et al. 2020).

$$\min_{f_{\boldsymbol{\theta}}} W_p^p\left(\left\{f_{\boldsymbol{\theta}}(\boldsymbol{z}_i)\right\}_{i=1}^K, \left\{\boldsymbol{x}_j\right\}_{j=1}^K\right)$$

$\{\boldsymbol{z}_i\}$ some random vectors, $\{\boldsymbol{x}_j\}$ some samples from the target distribution.

3. Conclusion

Take Home Message

- A powerful tool, well theoretically grounded, for manipulating distributions in machine learning.
- Despite its initial computational complexity, a lot of applications, even in large scale/deep learning settings.
- Others OT aspects (out the scope of the presentation): unbalanced OT, Gromov-Wasserstein distance (working with structured data), and many more

Some References

- G. Peyré and M. Cuturi,
  Computational Optimal Transport with Applications to Data Sciences, 2019
- N. country, R. Flamary, D. Tuia and A. Rakotomamonjy.
  Optimal Transport for Domain Adaptation, *PAMI 2017*
- R. Flamary et al. POT: Python Optimal Transport Library, 2017.

POT: Python Optimal Transport

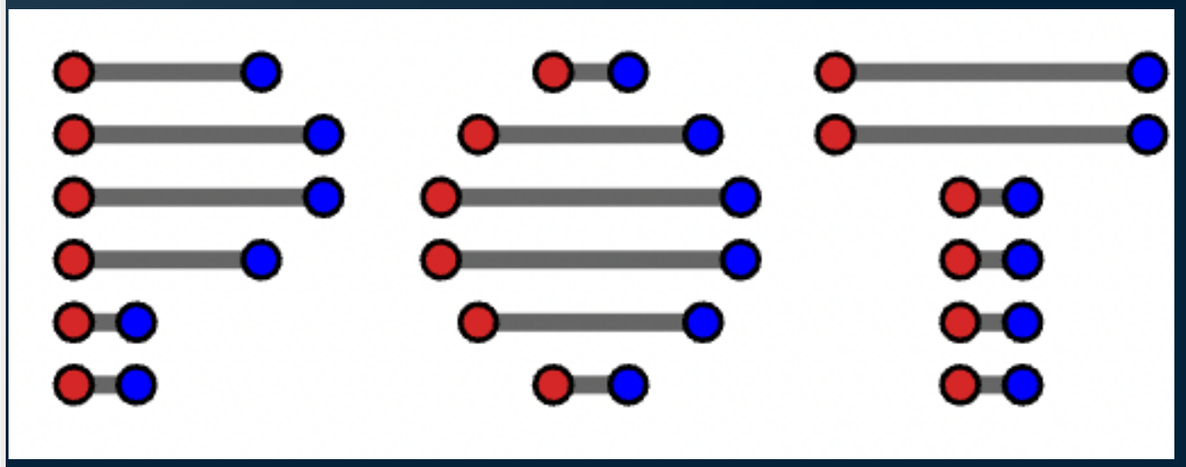
Contents

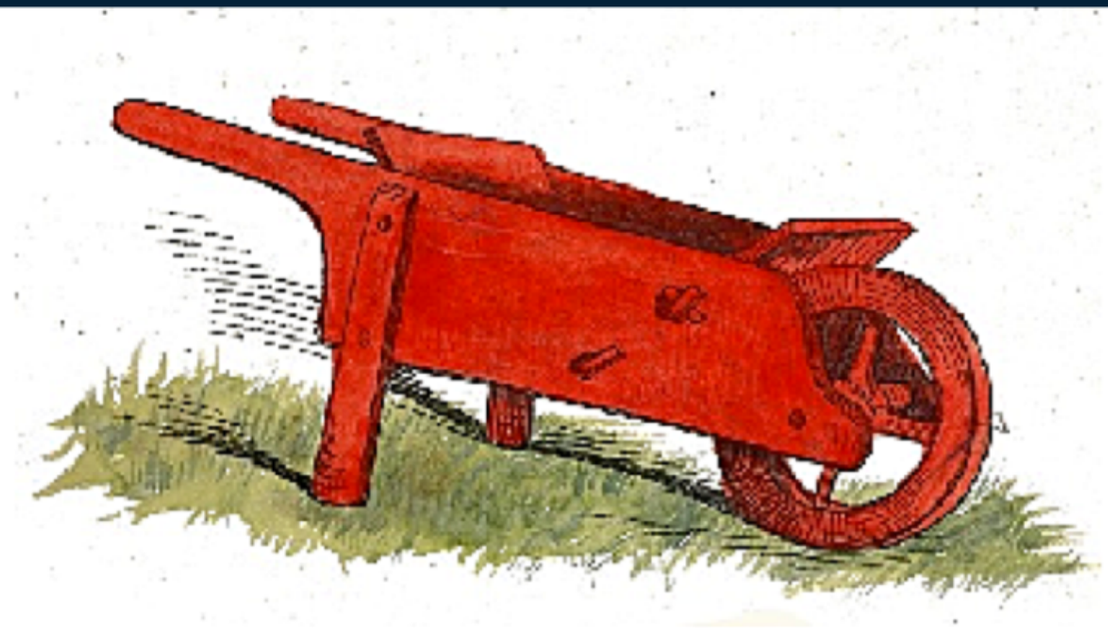
pypi package 0.7.0 | Anaconda Cloud 0.7.0 | build passing | codecov 92% | downloads 177k
downloads 86k total | license MIT

This open source Python library provide several solvers for optimization problems related to Optimal Transport for signal, image processing and machine learning.

Website and documentation: https://PythonOT.github.io/

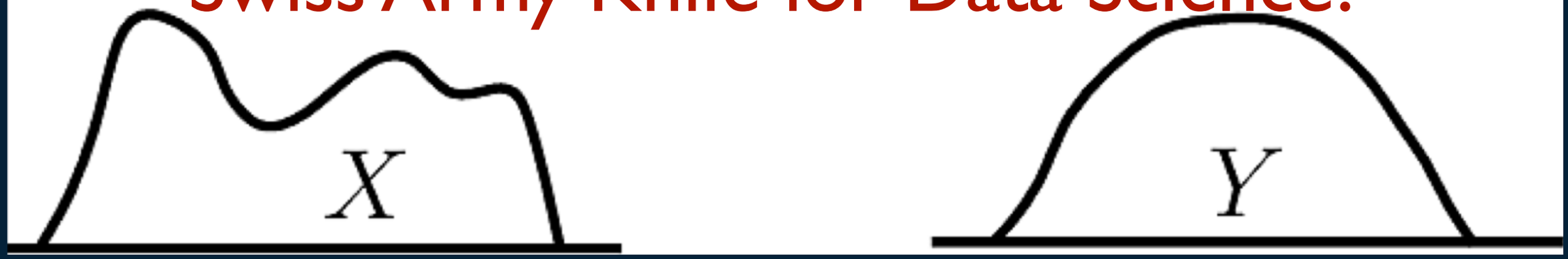
Source Code (MIT): https://github.com/PythonOT/POT





Optimal Transport

Swiss Army Knife for Data Science!



Thank You!