Introduction on Optimal Transport for Deep Learning

First definitions and properties

Kilian Fatras
March 24th, 2022

Mila, McGill
Introduction on deep learning and optimal transport

- Introduction on deep learning
  - Neural networks
  - Applications
  - Probability distributions

- Introduction on optimal transport
  - Definitions
  - Properties
  - Entropic variant
Introduction on Neural networks
Deep learning is a tool to estimate non-linear complex functions

- Neural networks: many stacked layers and each layer is made of neurones
- Parameters of neural networks: connections between layers
- Different layers: convolutional layers, fully connected layers, ...
Motivating example: Classification

- Find a function $f_\theta$ which describes the relationship between the space of images and the space of classes
- $f_\theta$ is a **neural network**!

$$f_\theta \left( \begin{array} \end{array} \right) = \begin{pmatrix} 0.1 \\
0.2 \\
0.7 \
\end{pmatrix}$$

- $n$ training samples: $(x_1, y_1), \ldots, (x_n, y_n)$
- Goal: minimizing the *empirical risk* with respect to $\theta$

$$\min_\theta R(f_\theta) = \min_\theta \frac{1}{N} \sum_{i=1}^{N} L(y_i, f_\theta(x_i))$$
Motivating example: Domain adaptation

Domain adaptation (DA) setting

- Two domains with same classes, only one with labels
- Goal: classify unlabeled target data with source labeled data

\[ x^s_i, x^t_j \text{ have same class } \rightarrow g_\phi(x_i) \approx g_\phi(x_j) \text{ and } y_i = f_\theta(g_\phi(x_j)) \]
Motivating example: Generative adversarial networks

Goal: generating new images

- Generative adversarial networks (GANs) developed in [Goodfellow et al., 2014]
- $G_{\theta}$ tries to fool $D_{\phi}$
- $D_{\phi}$ tries to predict if an image is real or not
Motivating example: Generative adversarial networks

Goal: generating new images

\[ P(X), P(Z) \] are probability distributions

Loss: \( \min \alpha \max \phi \mathbb{E}_{x \sim \alpha} \log (D_\phi(x)) - \mathbb{E}_{z \sim \zeta} \log (1 - D_\phi(G_\theta(z))) \)

The loss can be reformulated with a Jensen-Shannon divergence between generated and training distributions
Training samples as distributions paradigm

Applications use probability distributions to train neural networks

- Classification: function $L$ takes probability vectors as inputs
- Domain adaptation: align embedding probability distributions from domains
- GANs: distance between generated and training distributions

\[
\hat{\theta} = \arg\min_{\theta \in \Theta} L(\alpha_n, \beta_\theta)
\]

Goal: Find a suitable function $L$ between probability distributions
Divergence and metric between probability distributions

**Definition (Divergence)**

Consider a set $S$. A divergence on $S$ is a function $d : S \times S \mapsto [0, \infty]$ such that for all $x, y$:

- $d(x, y) \geq 0$ (non negativity)
- $d(x, y) = 0$ if and only if $x = y$ (separability)

**Definition (Distance/Metric)**

Consider a set $S$. A distance on $S$ is a function $d : S \times S \mapsto [0, \infty]$ such that for all $x, y, z$:

- $d(x, y) \geq 0$ (non negativity)
- $d(x, y) = 0$ if and only if $x = y$ (separability)
- $d(x, y) = d(y, x)$ (symmetry)
- $d(x, z) \leq d(x, y) + d(y, z)$ (triangle inequality)
Comparing probability distributions
Comparing probability distributions

Suppose $\varphi$ convex, $\varphi(1) = 0$ and $\alpha$ absolutely continuous wrt $\beta$. $\varphi$-divergences compare mass ratio point-wise $\alpha(x)/\beta(x)$ ($\beta(x) > 0$).

$$L_\varphi(\alpha|\beta) = \int_{\mathcal{X}} \varphi \left( \frac{d\alpha}{d\beta} \right) d\beta$$

We give several examples of $\varphi$-divergences.
Comparing probability distributions

Suppose \( \varphi \) convex, \( \varphi(1) = 0 \) and \( \alpha \) absolutely continuous \( \text{wrt} \ \beta \). \( \varphi \)-divergences compare mass ratio point-wise \( \alpha(x)/\beta(x) \) \( (\beta(x) > 0) \).

\[
L_\varphi(\alpha|\beta) = \int_X \varphi \left( \frac{d\alpha}{d\beta} \right) d\beta
\]

We can get the Kullback-Leibler divergence for \( \varphi(x) = x \log(x) \),

\[
KL(\alpha|\beta) = \int_X \log \left( \frac{d\alpha}{d\beta} \right) d\alpha
\]
Comparing probability distributions

Suppose $\varphi$ convex, $\varphi(1) = 0$ and $\alpha$ absolutely continuous \textit{wrt} $\beta$. $\varphi$-divergences compare mass ratio point-wise $\alpha(x)/\beta(x)$ ($\beta(x) > 0$).

$$L_\varphi(\alpha|\beta) = \int_{\mathcal{X}} \varphi \left( \frac{d\alpha}{d\beta} \right) d\beta$$

We can get the Total-Variation norm for $\varphi(x) = \frac{1}{2} |x - 1|$, 

$$TV(\alpha|\beta) = \int_{\mathcal{X}} \frac{1}{2} \left| \frac{d\alpha}{d\beta} - 1 \right| d\alpha$$
Comparing probability distributions

Suppose $\varphi$ convex, $\varphi(1) = 0$ and $\alpha$ absolutely continuous wrt $\beta$. $\varphi$-divergences compare mass ratio point-wise $\alpha(x)/\beta(x)$ ($\beta(x) > 0$).

$$L_\varphi(\alpha|\beta) = \int_X \varphi \left( \frac{d\alpha}{d\beta} \right) d\beta$$

- $\varphi$-divergences cannot compare Diracs
  - fail to capture the geometry
- $KL(\alpha|\beta_t) = +\infty$
Comparing probability distributions

Suppose $\varphi$ convex, $\varphi(1) = 0$ and $\alpha$ absolutely continuous wrt $\beta$. $\varphi$-divergences compare mass ratio point-wise $\alpha(x)/\beta(x)$ ($\beta(x) > 0$).

$$L_\varphi(\alpha|\beta) = \int_X \varphi \left( \frac{d\alpha}{d\beta} \right) d\beta$$

- $\varphi$-divergences cannot compare Diracs
  $\rightarrow$ fail to capture the geometry
- $\text{KL}(\alpha|\beta_t) = +\infty$ but $\text{KL}(\alpha|\beta_\infty) = 0$
Weak convergence topology

**Definition (Convergence in metric space)**
A sequence \( \{l_t\}_{t \in \mathbb{N}} \) of elements of a metric space \((S, d)\) is said to converge to a limit \( l \in S \) if \( \lim_{t \to \infty} d(l_t, l) = 0 \).

For probability distributions, sequence \( \beta_t \) converges to \( \beta \) with respect to a divergence \( d \) if \( \lim_{t \to \infty} d(\beta_t, \beta) = 0 \). (Be careful with the symmetry!)

\( \varphi \)-divergences do not metrize the weak convergence.

**Example (TV-divergences)**
For the probability sequence \( \delta_{\frac{1}{n}} \), It is clear that \( \lim_{t \to \infty} \delta_{\frac{1}{n}} = \delta_0 \) but we have \( \lim_{t \to \infty} \text{TV}(\delta_{\frac{1}{n}}, \delta_0) = \lim_{t \to \infty} 1 = 1 \).

So we are looking for a function \( d \) which can compare probability distributions and which metrizes the weak convergence.
Introduction on Optimal Transport
Optimal Transport definition

**Ingredients**

- Probability distributions $\alpha \in \mathcal{P}(\mathcal{X})$ and $\beta \in \mathcal{P}(\mathcal{Y})$
- A ground cost $c : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^+$ with $\mathcal{X}$ and $\mathcal{Y}$ metric spaces
Optimal Transport definition

**Ingredients**

- Probability distributions $\alpha \in \mathcal{P}(\mathcal{X})$ and $\beta \in \mathcal{P}(\mathcal{Y})$
- A ground cost $c : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^+$ with $\mathcal{X}$ and $\mathcal{Y}$ metric spaces
Definition (Kantorovich problem [Kantorovich, 1942])

\[ \min_{\pi \in U(\alpha, \beta)} \int_{X \times Y} c(x, y) d\pi(x, y) \]

with: \( U(\alpha, \beta) = \{ \pi \in \mathcal{P}(X \times Y), \int_Y \pi(x, y) dy = \alpha, \int_X \pi(x, y) dx = \beta \} \)
Discrete Optimal Transport

Discrete ingredients

- Discrete distributions \( \alpha = \sum_{i=1}^{n} a_i \delta_{x_i} \) and \( \beta = \sum_{j=1}^{n} b_j \delta_{y_j} \)
- Cost matrix \( C = C(X, Y) \), such that \( C_{i,j} = c(x_i, y_j) \)
For discrete distributions, OT becomes a linear program:

**Definition (Discrete Optimal Transport)**

\[
\text{OT}(\alpha, \beta, C) = \min_{\Pi \in U(a, b)} \sum_{i, j} \Pi_{i, j} C_{i, j}
\]

\[
U(a, b) = \left\{ \Pi \in (\mathbb{R}^+)^{n_1 \times n_2} \mid \Pi 1_{n_1} = a, \Pi^T 1_{n_2} = b \right\}
\]
Consider the following 2D example:

The probability distribution weights are:

\[ a = \left[ \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4} \right]^\top \]

\[ b = \left[ \frac{1}{3}, \frac{2}{3} \right]^\top \]

What is the optimal transport plan \( \Pi \)?
Example of optimal plan

OT between two 2D discrete distributions

\[ a_1 = \frac{1}{4}, \quad a_2 = \frac{1}{4}, \quad a_3 = \frac{1}{4}, \quad a_4 = \frac{1}{4}, \quad b_1 = \frac{1}{3}, \quad b_2 = \frac{2}{3} \]

\[
\Pi = \begin{bmatrix}
0.083 & 0.167 \\
0 & 0.25 \\
0.25 & 0 \\
0 & 0.25 \\
\end{bmatrix}
\quad \Pi_1 2 = \begin{bmatrix}
0.083 & 0.167 \\
0 & 0.25 \\
0.25 & 0 \\
0 & 0.25 \\
\end{bmatrix}
\begin{bmatrix}
1 \\
1 \\
1 \\
1 \\
\end{bmatrix} = \begin{bmatrix}
\frac{1}{4} \\
\frac{1}{4} \\
\frac{1}{4} \\
\frac{1}{4} \\
\end{bmatrix} = \alpha
\]
Example of optimal plan

OT between two 2D discrete distributions

Source samples
Target samples

\(\begin{align*}
\Pi &= \begin{bmatrix}
0.083 & 0.167 \\
0 & 0.25 \\
0.25 & 0 \\
0 & 0.25 \\
\end{bmatrix} \\
\Pi^\top \mathbf{1}_2 &= \begin{bmatrix}
0.083 & 0 & 0.25 & 0 \\
0.167 & 0.25 & 0 & 0.25 \\
1 & 1 & 1 & 1 \\
\end{bmatrix} \begin{bmatrix}
1 \\
1 \\
2/3 \\
\end{bmatrix} = \begin{bmatrix}
1/3 \\
2/3 \\
\end{bmatrix} = \mathbf{b}
\end{align*}\)
Optimal Transport connections

Computed with Python optimal Transport! [Flamary et al., 2021]
Optimal Transport connections

Euclidean cost  
Squared Euclidean cost  
Sqrt Euclidean cost

Computed with Python optimal Transport! [Flamary et al., 2021]
Wasserstein distance

Some properties

- Leverages geometry of sample spaces through $C$
- A solution always exists (ex. $\pi = \alpha \otimes \beta$)
- $\langle \Pi, C \rangle_F$ is linear in the transport plan and in the cost
- Convex in the transport plan $\Pi$
Wasserstein distance

Some properties

- Leverages geometry of sample spaces through $C$
- A solution always exists (ex. $\pi = \alpha \otimes \beta$)
- $\langle \Pi, C \rangle_F$ is linear in the transport plan and in the cost
- Convex in the transport plan $\Pi$

Definition (Wasserstein distance)

$C$ is a ground metric, then OT cost $W_p$ is a metric for $p \geq 1$ and where

$$W_p(\alpha, \beta, C^p) = \left( \min_{\Pi \in U(a,b)} \langle \Pi, C^p \rangle_F \right)^{1/p}$$

Proposition (Weak convergence)

The Wasserstein distance metrizes the weak convergence.

$$W_p(\delta_{\frac{1}{n}}, \delta_0, c) = c(\delta_{\frac{1}{n}}, \delta_0)$$
Optimal Transport has a dual program:

**Proposition (Kantorovich duality)**

\[ \mathcal{L}(\alpha, \beta, c) = \sup_{(f,g)\in\mathcal{R}(c)} \int_X f(x) d\alpha(x) + \int_Y g(y) d\beta(y). \]

Where the set of admissible dual potentials is:

\[ \mathcal{R}(c) = \{(f,g) \in \mathcal{C}(X) \times \mathcal{C}(Y) : \forall (x,y), f(x) + g(y) \leq c(x,y)\}. \]

**Proposition (Discrete Kantorovich duality)**

\[ \mathcal{L}(\alpha, \beta, C) = \max_{(f,g)\in\mathcal{R}(C)} \langle f, a \rangle + \langle g, b \rangle. \]

Where the set of admissible dual potentials is:

\[ \mathcal{R}(C) = \{(f,g) \in \mathbb{R}^n \times \mathbb{R}^n : \forall (i,j) \in [n]^2, f_i + g_j \leq C_{i,j}\}. \]

Can be solved with simplex algorithm with complexity of \( \mathcal{O}(n^3 \log(n)) \).
For the case of the Wasserstein-1 distance, we have:

**Proposition (Kantorovich–Rubinstein duality)**

\[
W_1(\alpha, \beta, C) = \sup_{f \in \text{Lip}^1(X)} \mathbb{E}_{x \sim \alpha}[f(x)] - \mathbb{E}_{z \sim \beta}[f(z)].
\]

Supremum is intractable → approximate it with a neural network.

Suppose \(\alpha\) is the probability distributions of real images and \(\beta_\theta\) is a parametric distribution we want to fit to \(\alpha\). We want to minimize

\[
\min_{\theta \in \Theta} W_1(\alpha, \beta_\theta, C) = \min_{\theta \in \Theta} \sup_{f \in \text{Lip}^1(X)} \mathbb{E}_{x \sim \alpha}[f(x)] - \mathbb{E}_{z \sim \beta}[f(z)],
\]

\[
\approx \min_{\theta \in \Theta} \max_{\phi \in \Phi} \mathbb{E}_{x \sim \alpha}[f_\phi(x)] - \mathbb{E}_{z \sim \beta}[f_\phi(z)].
\]

Where \(\Phi\) is compact. To ensure Lipschitz constraint WGAN clips weights and WGAN-GP uses a gradient penalty.
Summary on neural networks and optimal transport

- **Summary on neural networks**
  - Neural networks are stacked layers of neurons
  - Competitive methods on classification, domain adaptation and GANs

- **Summary on optimal transport**
  + Loss function/distance between distributions of samples
  + Leverages geometry of sample spaces through $C$
  - Cubical computational complexity of discrete OT
  + Useful dual formulations
Entropic Optimal Transport
Entropic Optimal Transport

Definition (Entropic Optimal Transport [Cuturi, 2013])

\[
\text{OT}^\varepsilon(\alpha, \beta, C) = \min_{\Pi \in U(a, b)} \sum_{i,j} \Pi_{i,j} C_{i,j} + \varepsilon \text{KL}(\Pi|a \otimes b)
\]

\[
\forall x, y \in \mathbb{R}_+^n, \text{KL}(x|y) = \sum_i x_i \log \left( \frac{x_i}{y_i} \right) - x_i + y_i
\]

- Functional is strongly convex in the transport plan
- Computational complexity of entropic OT is \(\mathcal{O}\left(\frac{n^2}{\varepsilon}\right)\)
Example of optimal plan

E-OT between two 2D discrete distributions

\[ a_1 = \frac{1}{4}, \quad a_2 = \frac{1}{4}, \quad a_3 = \frac{1}{4}, \quad a_4 = \frac{1}{4}, \quad b_1 = \frac{1}{3}, \quad b_2 = \frac{2}{3} \]

E-OT plan, \( \varepsilon = 1 \)

\[
\Pi = \begin{bmatrix}
0.10 & 0.15 \\
0.02 & 0.23 \\
0.16 & 0.09 \\
0.05 & 0.20
\end{bmatrix} \quad \Pi 1_2 = \begin{bmatrix}
0.10 & 0.15 \\
0.02 & 0.23 \\
0.16 & 0.09 \\
0.05 & 0.20
\end{bmatrix} \begin{bmatrix}
1 \\
1 \\
1 \\
1
\end{bmatrix} = \begin{bmatrix}
\frac{1}{4} \\
\frac{1}{4} \\
\frac{1}{4} \\
\frac{1}{4}
\end{bmatrix} = a
\]
Example of optimal plan

\[ \Pi = \begin{bmatrix} 0.10 & 0.15 \\ 0.02 & 0.23 \\ 0.16 & 0.09 \\ 0.05 & 0.20 \end{bmatrix} \]

\[ \Pi^\top \mathbf{1}_2 = \begin{bmatrix} 0.10 & 0.02 & 0.16 & 0.05 \\ 0.15 & 0.23 & 0.09 & 0.20 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1/3 \\ 2/3 \end{bmatrix} = b \]
Proposition (Convergence with $\varepsilon$)

We denote $\Pi^\varepsilon$ the optimal transport plan of entropic OT. We have the following convergence property:

$$\text{OT}^\varepsilon(\alpha, \beta, C) \xrightarrow[\varepsilon \to 0]{} \text{OT}(\alpha, \beta, C)$$

$$\Pi^\varepsilon \xrightarrow[\varepsilon \to +\infty]{} a \otimes b$$

Proposition (Solution of the regularized Kantarovich problem)

The solution of the regularized (entropic) Kantarovich problem has the form:

$$\forall (i, j) \in [n] \times [m], P^\varepsilon_{i,j} = u_i \exp(-C/\varepsilon)_{i,j} v_j$$

for 2 unknown scaling variable $(u, v) \in \mathbb{R}_+^n \times \mathbb{R}_+^m$. 
The algorithm performs alternatively a scaling along the rows and columns of $K$ to match the desired marginals.

- Computational complexity $\mathcal{O}(\kappa n^2)$
- Fast implementation in parallel (GPU)
Optimal Transport connections

Computed with Python optimal Transport ! [Flamary et al., 2021]
Optimal Transport connections

Euclidean, $\lambda = 0.005$  
Squared Euclidean, $\lambda = 0.005$  
Sqrt Euclidean, $\lambda = 0.005$

Computed with Python optimal Transport ! [Flamary et al., 2021]
Optimal Transport has a dual program:

**Proposition (entropic OT duality)**

\[
\text{OT}^\varepsilon(\alpha, \beta, C) = \max_{(f,g) \in (\mathbb{R}^n)^2} \langle f, a \rangle + \langle g, b \rangle - \varepsilon \langle \frac{e^f}{\varepsilon}, K \frac{e^g}{\varepsilon} \rangle.
\]

Note the unconstrained dual contrary to exact OT.

The optimal \((f, g)\) are linked to scalings \((u, v)\) appearing in the Sinkhorn algorithm through

\[
(u, v) = \left( \frac{e^f}{\varepsilon}, \frac{e^g}{\varepsilon} \right)
\]

(1)
Proposition (Derivative with respect to weights)

For $\varepsilon > 0$, $(a, b) \mapsto \text{OT}^\varepsilon((a, X), (b, Y), C')$ is differentiable. Its gradient reads

$$\nabla \text{OT}^\varepsilon((a, X), (b, Y), C') = (f, g)$$

where $(f, g)$ is the unique solution, centered such that $\sum_i f_i = \sum_j g_j = 0$. For $\varepsilon = 0$, this formula defines the elements of the sub-differential.

Proposition (Derivative with respect to the cost)

For fixed input histograms $(a, b)$, for $\varepsilon > 0$, the mapping $C \mapsto \text{OT}^\varepsilon((a, X), (b, Y), C')$ is smooth, and

$$\nabla_C \text{OT}^\varepsilon((a, X), (b, Y), C') = \Pi^\varepsilon$$

For $\varepsilon = 0$, this formula defines the set of upper gradients.
Limits of entropic optimal transport

Unfortunately, entropic OT is not a distance.

**Proposition (Entropic OT losses distance properties)**

\[ OT^\varepsilon(\alpha, \alpha, C) > 0. \]

We can nonetheless define a new loss function called the Sinkhorn divergence as:

**Proposition (Sinkhorn divergences)**

\[
S^\varepsilon(\alpha, \beta, C) = OT^\varepsilon(\alpha, \beta, C) - \frac{1}{2}(OT^\varepsilon(\alpha, \alpha, C) + OT^\varepsilon(\beta, \beta, C)).
\]

The Sinkhorn divergence defines a divergence between probability measures [Feydy et al., 2019] and interpolate between OT and MMD [Gretton et al., 2012]. It has also better statistical properties than OT.
Unbalanced Optimal Transport
Unbalanced Optimal Transport

**Definition**

Unbalanced Optimal transport measures the distance between distributions, but with relaxed marginals.

\[
\text{UOT}^\tau,\varepsilon (\alpha, \beta, c) = \min_{\pi \in M_+(X \times Y)} \int c d\pi + \tau (\text{KL}(\pi_1||\alpha) + \text{KL}(\pi_2||\beta)),
\]

where \( \pi \) is the transport plan, \( \pi_1 \) and \( \pi_2 \) the plan’s marginals, \( \tau \geq 0 \) is the marginal penalization and \( \varepsilon \geq 0 \) is the regularization coefficient.

**Difference with OT**

- \( \pi \in U(\alpha, \beta) \rightarrow \pi \in M_+(X \times Y) \)
- Fixed marginal constraints are replaced by \( \text{KL}(\pi_1||\alpha) \) penalties
- Unique marginals \( \pi_1 \) and \( \pi_2 \)
- \( \text{KL} \) can be replaced by \( \text{TV} \)
Entropic unbalanced Optimal Transport

**Definition**

Entropic unbalanced Optimal transport measures the distance between distributions, but with relaxed marginals.

$$UOT^{\tau,\varepsilon}(\alpha, \beta, c) = \min_{\pi \in \mathcal{M}_+(X \times Y)} \int c d\pi + \varepsilon KL(\pi | \alpha \otimes \beta)$$

$$+ \tau (KL(\pi_1 || \alpha) + KL(\pi_2 || \beta)),$$

where $\pi$ is the transport plan, $\pi_1$ and $\pi_2$ the plan’s marginals, $\tau \geq 0$ is the marginal penalization and $\varepsilon \geq 0$ is the regularization coefficient.

**Difference with UOT**

- Unique solution $\Pi$
- Can be solved with a generalized Sinkhorn algorithm
- $UOT^{\tau,\varepsilon}(\alpha, \alpha, c) > 0$ but can define a Sinkhorn UOT variant

[Séjourné et al., 2019]
Influence of $\tau$

Let us study the optimal transport plan for a fixed problem and a various $\tau$.

Key message: Smaller $\tau$ decreases the transported mass as it is less costly to be "lazy".
Influence of higher cost

Let us study the optimal transport plan for a dynamic problem and a fixed $\tau$.

Key message: The more costly a sample is to transport, the less it is transported.
Time experiment

Limits
Can not be used in Big Data scenario!
Minibatch Optimal Transport
Let $m \leq n$, [Damodaran et al., 2018, Genevay et al., 2018] compute optimal transport between minibatch of distributions.

**Minibatch strategy**

- Select $m$ samples without replacement at random in domains
- Compute OT between the minibatches
- Average several MBOT terms → complexity $O(m^3)$
Minibatch Optimal Transport definition

Expectation of minibatches

Computing OT kernel $h$ between minibatches estimates:

$$E_h(\alpha, \beta, C) := \mathbb{E}_{(X,Y) \sim \alpha \otimes m \otimes \beta \otimes m} [h(\mu_m, \mu_m, C(X, Y))]$$

- Can be defined for OT variants $h$
- Justified in [Fatras et al., 2020]
Estimate minibatch OT distance

**Definition (Complete minibatch estimator)**

\[
\bar{h}^m(X, Y) := \left( \begin{pmatrix} n \\ m \end{pmatrix} \right)^{-2} \sum_{I,J \in \mathcal{P}_m} h(\mu_m, \mu_m, C_{I,J})
\]

\[
\Pi^m(X, Y) := \left( \begin{pmatrix} n \\ m \end{pmatrix} \right)^{-2} \sum_{I,J \in \mathcal{P}_m} \Pi_{I,J}
\]

- where \( \mathcal{P}_m \) is the set of all \( m \)-tuples without replacement
- \( \Pi^m(X, Y) \) is an admissible transport plan between the input probability distributions \( \Pi \in U(\mu_n, \mu_n) \)

**Definition (Incomplete minibatch estimator)**

\[
\tilde{h}^m_k(X, Y) := k^{-1} \sum_{(I,J) \in D_k} h(\mu_m, \mu_m, C_{I,J})
\]

where \( k > 0 \) is an integer and \( D_k \) is a set of cardinality \( k \) whose elements are minibatches drawn at random
From the 1D OT closed-form formula, we have:

\[
\pi_{j,k} = \frac{1}{m} \left( \frac{n}{m} \right)^{-2} \sum_{i=i_{\text{min}}}^{i_{\text{max}}} \binom{j-1}{i-1} \binom{k-1}{i-1} \binom{n-j}{m-i} \binom{n-k}{m-i}
\]

where \(i_{\text{min}} = \max(0, m - n + j, m - n + k)\) and \(i_{\text{max}} = \min(j, k)\)
A few key home message on minibatch OT.

- Not a distance
- Can not define a divergence like Sinkhorn divergence
- Better statistical properties
- A new loss function based on OT but not OT
Applications
Generative models

Taken from [Gulrajani et al., 2017].
Office Home Domain Adaptation dataset

Network: pre-trained ResNet 50 with an additional classification layer.

Figure taken from [Venkateswara et al., 2017]. 65 classes in the source and target domains for balanced DA and 25 classes in the target domains for partial DA.
## Domain Adaptation experiments

<table>
<thead>
<tr>
<th>Method</th>
<th>A-C</th>
<th>A-P</th>
<th>A-R</th>
<th>C-A</th>
<th>C-P</th>
<th>C-R</th>
<th>P-A</th>
<th>P-C</th>
<th>P-R</th>
<th>R-A</th>
<th>R-C</th>
<th>R-P</th>
<th>avg</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DA</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESNET-50</td>
<td>34.9</td>
<td>50.0</td>
<td>58.0</td>
<td>37.4</td>
<td>41.9</td>
<td>46.2</td>
<td>38.5</td>
<td>31.2</td>
<td>60.4</td>
<td>53.9</td>
<td>41.2</td>
<td></td>
<td>59.9</td>
</tr>
<tr>
<td>DANN (*)</td>
<td>44.3</td>
<td>59.8</td>
<td>69.8</td>
<td>48.0</td>
<td>58.3</td>
<td>63.0</td>
<td>49.7</td>
<td>42.7</td>
<td>70.6</td>
<td>64.0</td>
<td>51.7</td>
<td></td>
<td>78.3</td>
</tr>
<tr>
<td>CDAN-E (*)</td>
<td>52.5</td>
<td>71.4</td>
<td>76.1</td>
<td>59.7</td>
<td>69.9</td>
<td>71.5</td>
<td>58.7</td>
<td>50.3</td>
<td>77.5</td>
<td>70.5</td>
<td>57.9</td>
<td></td>
<td><strong>83.5</strong></td>
</tr>
<tr>
<td>DEEPJDOT (*)</td>
<td>50.7</td>
<td>68.6</td>
<td>74.4</td>
<td>59.9</td>
<td>65.8</td>
<td>68.1</td>
<td>55.2</td>
<td>46.3</td>
<td>73.8</td>
<td>66.0</td>
<td>54.9</td>
<td></td>
<td>78.3</td>
</tr>
<tr>
<td>ALDA (*)</td>
<td>52.2</td>
<td>69.3</td>
<td>76.4</td>
<td>58.7</td>
<td>68.2</td>
<td>71.1</td>
<td>57.4</td>
<td>49.6</td>
<td>76.8</td>
<td>70.6</td>
<td>57.3</td>
<td></td>
<td>82.5</td>
</tr>
<tr>
<td>ROT (*)</td>
<td>47.2</td>
<td>71.8</td>
<td>76.4</td>
<td>58.6</td>
<td>68.1</td>
<td>70.2</td>
<td>56.5</td>
<td>45.0</td>
<td>75.8</td>
<td>69.4</td>
<td>52.1</td>
<td></td>
<td>80.6</td>
</tr>
<tr>
<td>JUMBOT</td>
<td><strong>55.2</strong></td>
<td><strong>75.5</strong></td>
<td><strong>80.8</strong></td>
<td><strong>65.5</strong></td>
<td><strong>74.4</strong></td>
<td><strong>74.9</strong></td>
<td><strong>65.2</strong></td>
<td><strong>52.7</strong></td>
<td><strong>79.2</strong></td>
<td><strong>73.0</strong></td>
<td><strong>59.9</strong></td>
<td></td>
<td><strong>83.4</strong></td>
</tr>
<tr>
<td><strong>PDA</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESNET-50</td>
<td>46.3</td>
<td>67.5</td>
<td>75.9</td>
<td>59.1</td>
<td>59.9</td>
<td>62.7</td>
<td>58.2</td>
<td>41.8</td>
<td>74.9</td>
<td>67.4</td>
<td>48.2</td>
<td></td>
<td>74.2</td>
</tr>
<tr>
<td>DEEPJDOT (*)</td>
<td>48.2</td>
<td>66.2</td>
<td>76.6</td>
<td>56.1</td>
<td>57.8</td>
<td>64.5</td>
<td>58.3</td>
<td>42.7</td>
<td>73.5</td>
<td>65.7</td>
<td>48.2</td>
<td></td>
<td>73.7</td>
</tr>
<tr>
<td>PADA</td>
<td>51.9</td>
<td>67.0</td>
<td>78.7</td>
<td>52.2</td>
<td>53.8</td>
<td>59.0</td>
<td>52.6</td>
<td>43.2</td>
<td>78.8</td>
<td>73.7</td>
<td>56.6</td>
<td></td>
<td>77.1</td>
</tr>
<tr>
<td>ETN</td>
<td>59.2</td>
<td>77.0</td>
<td>79.5</td>
<td>62.9</td>
<td>65.7</td>
<td>75.0</td>
<td>68.3</td>
<td>55.4</td>
<td>84.4</td>
<td>75.7</td>
<td>57.7</td>
<td></td>
<td><strong>84.5</strong></td>
</tr>
<tr>
<td>BA3US (*)</td>
<td>56.7</td>
<td>76.0</td>
<td><strong>84.8</strong></td>
<td>73.9</td>
<td>67.8</td>
<td><strong>83.7</strong></td>
<td>72.7</td>
<td>56.5</td>
<td>84.9</td>
<td>77.8</td>
<td>64.5</td>
<td></td>
<td>83.8</td>
</tr>
<tr>
<td>JUMBOT</td>
<td><strong>62.7</strong></td>
<td><strong>77.5</strong></td>
<td>84.4</td>
<td><strong>76.0</strong></td>
<td><strong>73.3</strong></td>
<td>80.5</td>
<td><strong>74.7</strong></td>
<td><strong>60.8</strong></td>
<td><strong>85.1</strong></td>
<td><strong>80.2</strong></td>
<td><strong>66.5</strong></td>
<td></td>
<td><strong>83.9</strong></td>
</tr>
</tbody>
</table>

OT have state-of-the-art results [Fatras et al., 2021].


Interpolating between optimal transport and mmd using sinkhorn divergences.
In *Proceedings of Machine Learning Research*.

**Pot: Python optimal transport.**

Learning generative models with sinkhorn divergences.
In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*.

**Generative adversarial nets.**
A kernel two-sample test.

Improved training of wasserstein gans.

On translation of mass (in russian).
_Proceedings of the USSR Academy of Sciences._

Sinkhorn divergences for unbalanced optimal transport.

Deep hashing network for unsupervised domain adaptation.