Amortized optimization for computing optimal transport maps

Brandon Amos • Meta AI (FAIR) NYC

http://github.com/bamos/presentations

The Kantorovich dual for optimal transport

Chapter 5 of *Optimal transport: old and new,* Villani.

Given measures α, β and a cost c, the **Kantorovich dual formulation** is

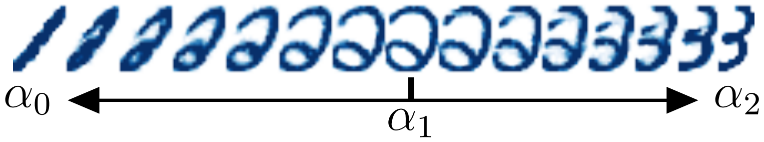
$$\hat{\psi}(\alpha,\beta,c) \in \operatorname*{argsup}_{\psi \in L^1(\alpha)} \int_{\mathcal{Y}} \psi^c(y)\,d\beta(y) - \int_{\mathcal{X}} \psi(x)\,d\alpha(x)$$

c-transform

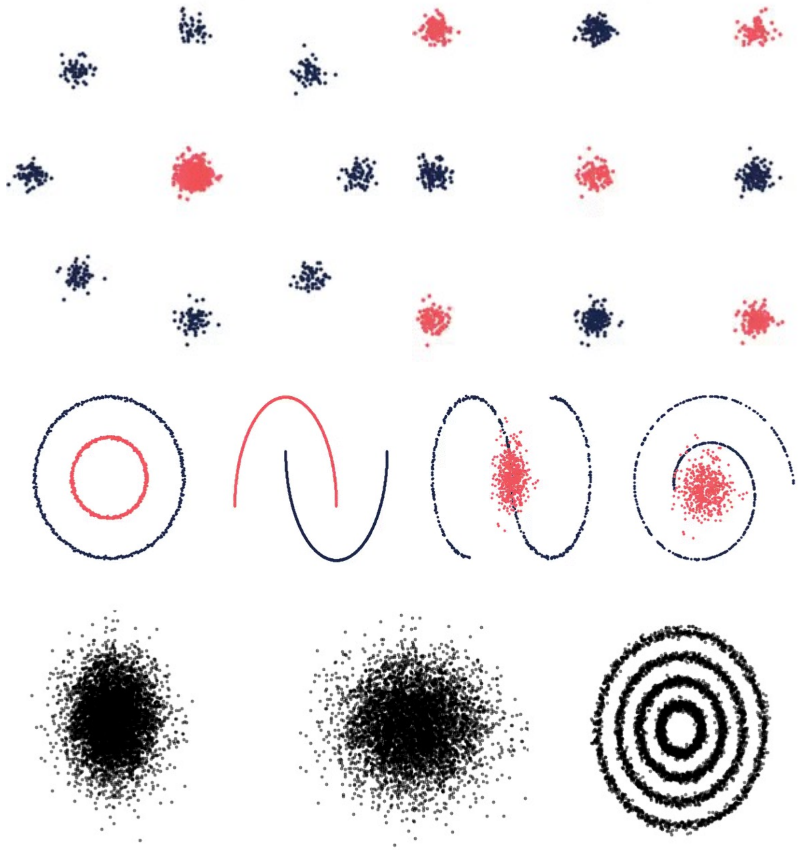
where $\quad \psi^c(y) \stackrel{\text{def}}{=} \inf_x \psi(x) + c(x,y)$

Many methods solve the dual:

- **Sinkhorn** for discrete measures (with entropy)
- **Euclidean Wasserstein-2 methods** (Brenier's theorem)



Meta Optimal Transport. Amos et al., 2022



On amortizing convex conjugates for optimal transport. Amos, 2022

Both optimization problems may be hard

Kantorovich dual

$$\hat{\psi}(\alpha, \beta, c) \in \underset{\psi \in L^1(\alpha)}{\operatorname{argsup}} \int_{\mathcal{Y}} \psi^c(y) d\beta(y) - \int_{\mathcal{X}} \psi(x) d\alpha(x)$$

Repeatedly solved for new measures and costs
Usually **solved from scratch** every time

***c*-transform**

$$\psi^c(y) \stackrel{\text{def}}{=} \inf_x \psi(x) + c(x, y)$$

Easy for small discrete measures ($\mathcal{X}$ finite)
Otherwise a **continuous optimization problem**
Repeatedly solved to evaluate the dual objective

Can machine learning help solve them? Yes!

Key idea of this talk: rapidly predict the solutions to these optimization problems
Leverages shared structure in the solution mapping

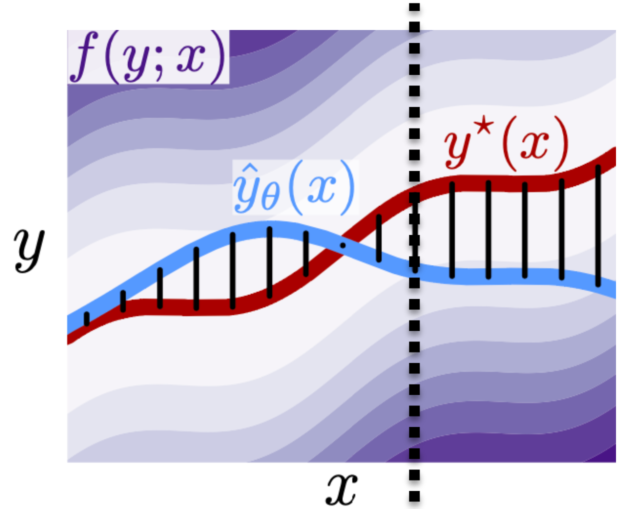
Amortized optimization

Tutorial on amortized optimization for learning to optimize over continuous domains. Amos, Foundations and Trends in Machine Learning (to appear)

Setup: Repeatedly solving continuous optimization problems of the form $y^\star(x) \in \operatorname{argmin}_y f(y; x)$
x is a **context** or **parameterization** of the optimization problem

Amortized optimization
Parameterize a **model** $\hat{y}_\theta(x)$
Optimize or learn to approximate the solution $\hat{y}_\theta(x) \approx y^\star(x)$

Amortization is widely deployed
Amortized variational inference (VAEs)
Meta-learning (hypernetworks, MAML)
Reinforcement learning (policy learning for actor-critic methods, SAC)



Vertical slices are optimization problems

Successes of amortization are **unconstrained continuous optimization problems**
Arises frequently in OT (Sinkhorn iterates, convex conjugate)
Makkuva et al. and Korotin et al. (W2GN) already using amortization

This talk: amortized optimization for OT

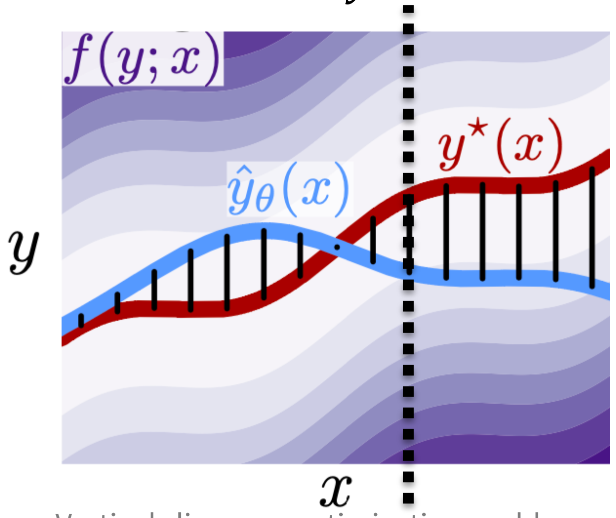
Amortizing the Kantorovich dual (Meta Optimal Transport)

$$\hat{\psi}(\alpha, \beta, c) \in \underset{\psi \in L^1(\alpha)}{\operatorname{argsup}} \int_{\mathcal{Y}} \psi^c(y) d\beta(y) - \int_{\mathcal{X}} \psi(x) d\alpha(x)$$

Amortizing the c-transform (the convex conjugate)

$$\psi^c(y) \stackrel{\text{def}}{=} \inf_x \psi(x) + c(x, y)$$

$$y^\star(x) \in \underset{y}{\operatorname{argmin}} f(y; x)$$



Vertical slices are optimization problems

Sinkhorn for entropic discrete OT

Primal formulation

$$P^\star(\alpha,\beta,c,\epsilon) \in \operatorname*{arg\,min}_{P\in U(a,b)} \langle C, P\rangle - \epsilon H(P)$$

$$H(P) := -\textstyle\sum_{i,j} P_{i,j}(\log(P_{i,j})-1)$$

(discrete entropy)

Algorithm 1 Sinkhorn($\alpha, \beta, c, \epsilon, f_0 = 0$)

for iteration i = 1 to N **do**
  $g_i \leftarrow \epsilon \log b - \epsilon \log\left(K^\top \exp\{f_{i-1}/\epsilon\}\right)$
  $f_i \leftarrow \epsilon \log a - \epsilon \log\left(K \exp\{g_i/\epsilon\}\right)$
end for
Compute P_N from f_N, g_N using eq. (6)
return $P_N \approx P^\star$

Dual formulation

$$f^\star, g^\star \in \operatorname*{arg\,max}_{f\in\mathbb{R}^n, g\in\mathbb{R}^m} \langle f, a\rangle + \langle g, b\rangle - \epsilon\left\langle \exp\{f/\epsilon\}, K\exp\{g/\epsilon\}\right\rangle, \quad K_{i,j} := \exp\{-C_{i,j}/\epsilon\},$$

Mapping from the dual solution to the primal

$$P^\star_{i,j}(\alpha,\beta,c,\epsilon) := \exp\{f^\star_i/\epsilon\} K_{i,j} \exp\{g^\star_j/\epsilon\}$$

Meta OT for Sinkhorn

Parameterize the potential $\hat{f}_\theta(\alpha, \beta, c)$, e.g., as an MLP

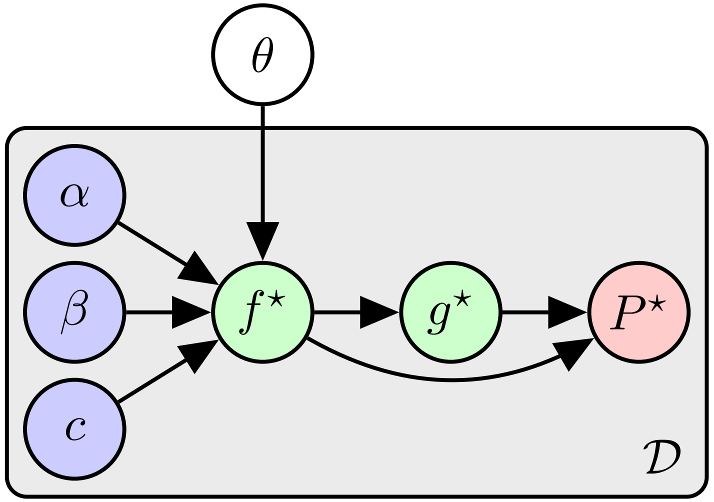
- Maps from the measures to the optimal duals

Learn the model

$$\min_\theta \mathop{\mathbb{E}}_{(\alpha,\beta,c)\sim\mathcal{D}} J(\hat{f}_\theta(\alpha,\beta,c); \alpha, \beta, c),$$

$$-J(f; \alpha, \beta, c) := \langle f, a\rangle + \langle g, b\rangle - \epsilon \left\langle \exp\{f/\epsilon\}, K \exp\{g/\epsilon\}\right\rangle$$

Prediction may be **inaccurate**, but not a problem
Can **check optimality** and **fine-tune with Sinkhorn**

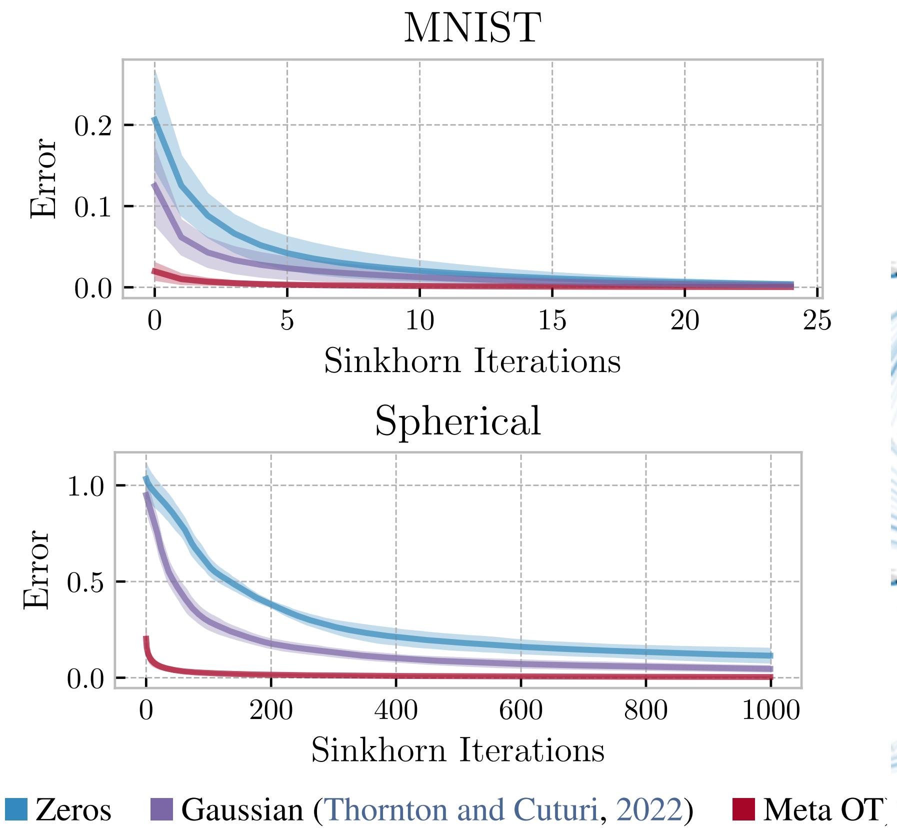


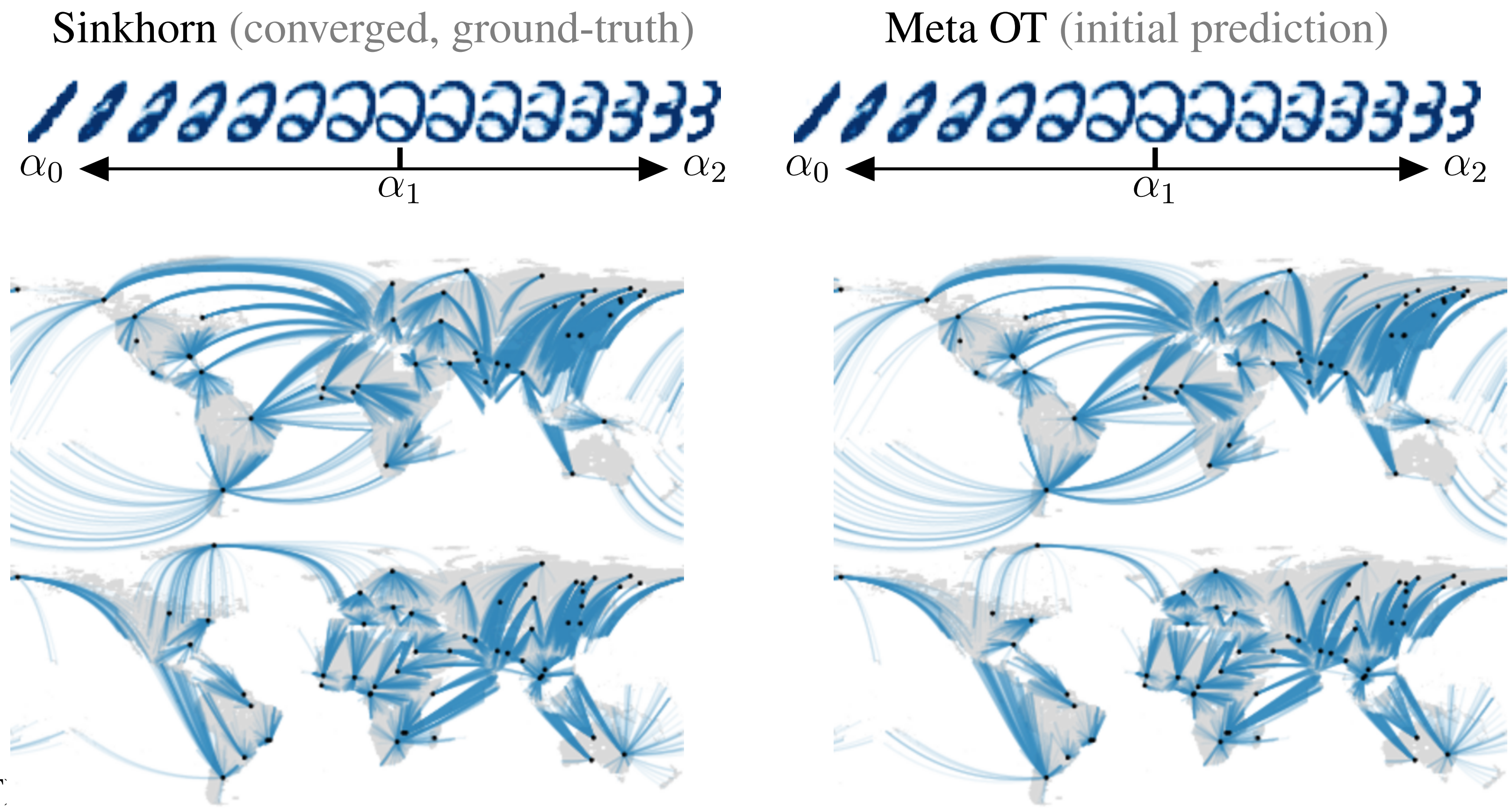
Discrete (Entropic)

Algorithm 1 Sinkhorn($\alpha, \beta, c, \epsilon, f_0 = 0$)

for iteration i = 1 to N **do**
  $g_i \leftarrow \epsilon \log b - \epsilon \log \left(K^\top \exp\{f_{i-1}/\epsilon\}\right)$
  $f_i \leftarrow \epsilon \log a - \epsilon \log \left(K \exp\{g_i/\epsilon\}\right)$
end for
Compute P_N from f_N, g_N using eq. (6)
return $P_N \approx P^\star$

Meta OT for Sinkhorn





Computing Euclidean Wasserstein-2 potentials

Wasserstein-2 Generative Networks. Korotin et al., ICLR 2020.

Primal formulation

$$W_2^2(\alpha,\beta) := \min_{\pi\in\mathcal{U}(\alpha,\beta)} \int_{\mathcal{X}\times\mathcal{Y}} \|x-y\|_2^2 \mathrm{d}\pi(x,y) = \min_T \int_{\mathcal{X}} \|x-T(x)\|_2^2 \mathrm{d}\alpha(x)$$

Dual formulation

$$\psi^\star(\,\cdot\,;\alpha,\beta) \in \operatorname*{arg\,min}_{\psi\in\text{convex}} \int_{\mathcal{X}} \psi(x)\mathrm{d}\alpha(x) + \int_{\mathcal{Y}} \overline{\psi}(y)\mathrm{d}\beta(y),$$

Loss for a parameterization of a potential ψ_φ

$$\mathcal{L}(\varphi) := \underbrace{\mathop{\mathbb{E}}_{x\sim\alpha}[\psi_\varphi(x)] + \mathop{\mathbb{E}}_{y\sim\beta}\left[\langle\nabla\overline{\psi_\varphi}(y), y\rangle - \psi_\varphi(\nabla\overline{\psi_\varphi}(y))\right]}_{\text{Cyclic monotone correlations (dual objective)}} + \gamma \underbrace{\mathop{\mathbb{E}}_{y\sim\beta}\|\nabla\psi_\varphi\circ\nabla\overline{\psi_\varphi}(y) - y\|_2^2}_{\text{Cycle-consistency regularizer}}, \quad (12)$$

Algorithm 2 W2GN(α,β,φ_0)

for iteration $i = 1$ to N **do**
  Sample from (α,β) and estimate $\mathcal{L}(\varphi_{i-1})$
  Update φ_i with approximation to $\nabla_\varphi\mathcal{L}(\varphi_{i-1})$
end for
return $T_N(\cdot) := \nabla_x\psi_{\varphi_N}(\cdot) \approx T^\star(\cdot)$

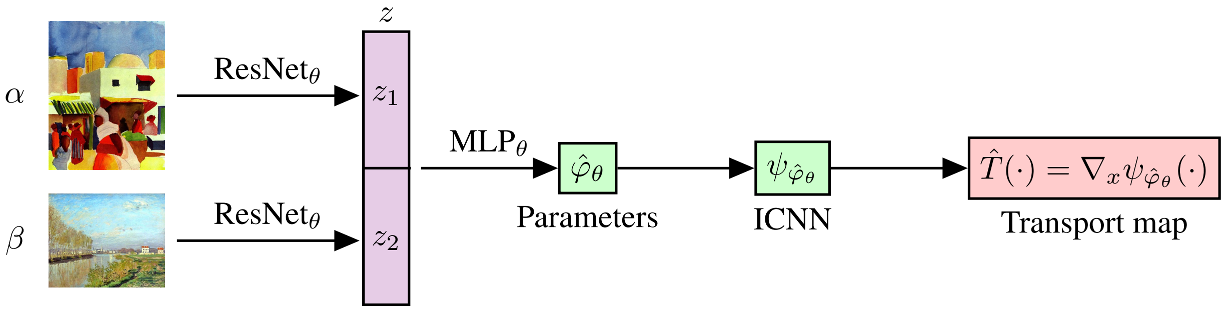
Brenier's theorem

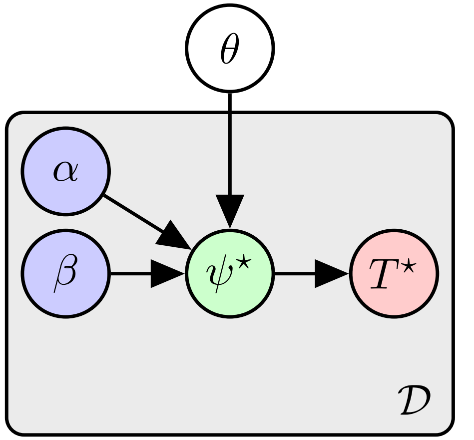
$$T^\star(x) = \nabla_x\psi^\star(x).$$

Meta OT for Euclidean Wasserstein-2 potentials

Parameterize the **model** $\hat{\varphi}_\theta(\alpha, \beta)$, e.g., a Meta ICNN

- Difference from continuous case, **dual potential is a function**
- **Hyper-network** mapping from the measures to the optimal dual **parameters**





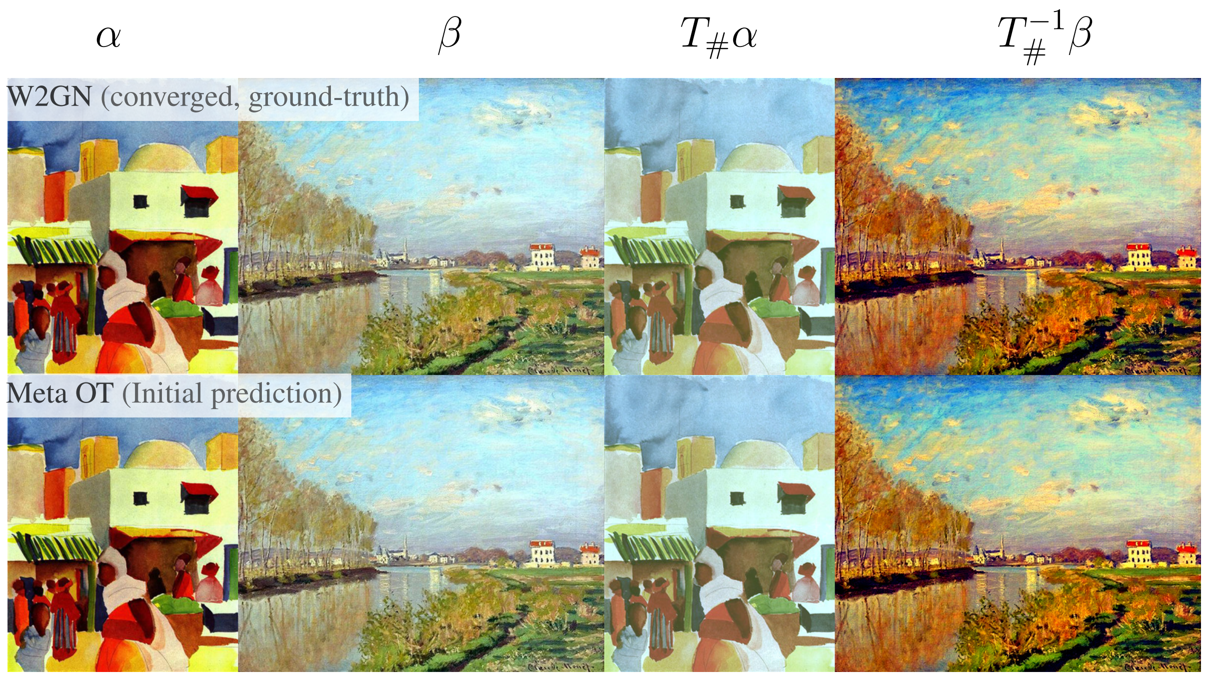
Continuous (Wasserstein-2)

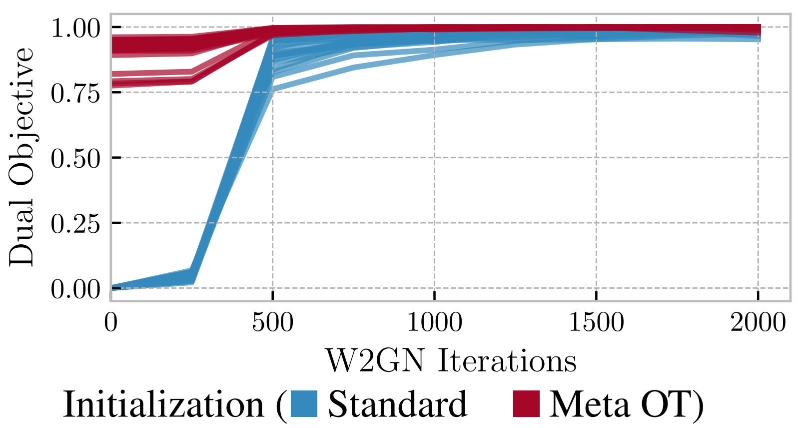
Learn the model with a meta version of the W2GN loss

$$\min_\theta \mathop{\mathbb{E}}_{(\alpha,\beta)\sim\mathcal{D}} \mathcal{L}(\hat{\varphi}_\theta(\alpha, \beta); \alpha, \beta).$$

$$\mathcal{L}(\varphi) := \underbrace{\mathop{\mathbb{E}}_{x\sim\alpha}[\psi_\varphi(x)] + \mathop{\mathbb{E}}_{y\sim\beta}\left[\langle \nabla\overline{\psi_\varphi}(y), y\rangle - \psi_\varphi(\nabla\overline{\psi_\varphi}(y))\right]}_{\text{Cyclic monotone correlations (dual objective)}} + \gamma \underbrace{\mathop{\mathbb{E}}_{y\sim\beta} \|\nabla\psi_\varphi \circ \nabla\overline{\psi_\varphi}(y) - y\|_2^2}_{\text{Cycle-consistency regularizer}}, \quad (12)$$

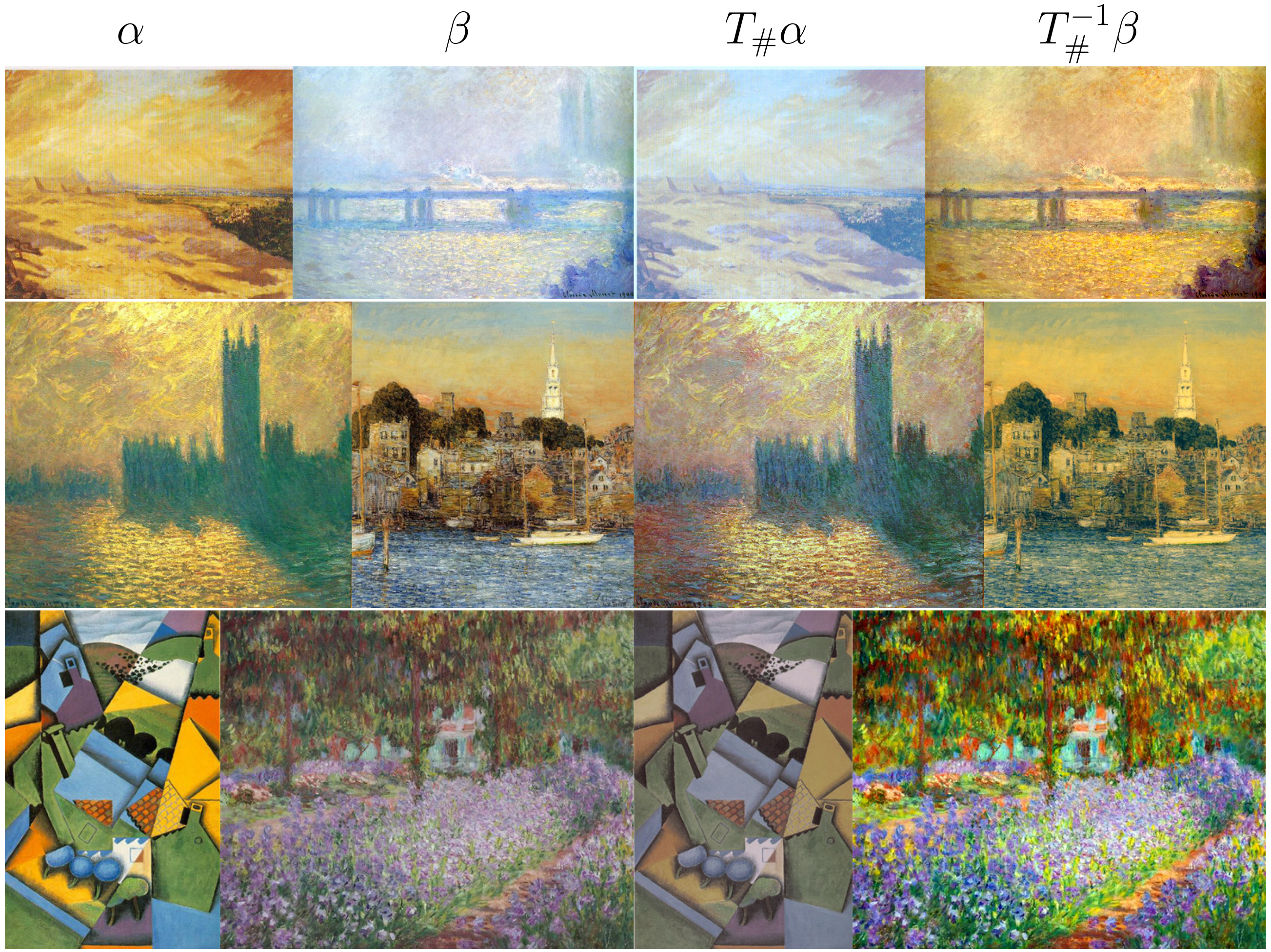
Continuous OT with Meta ICNNs

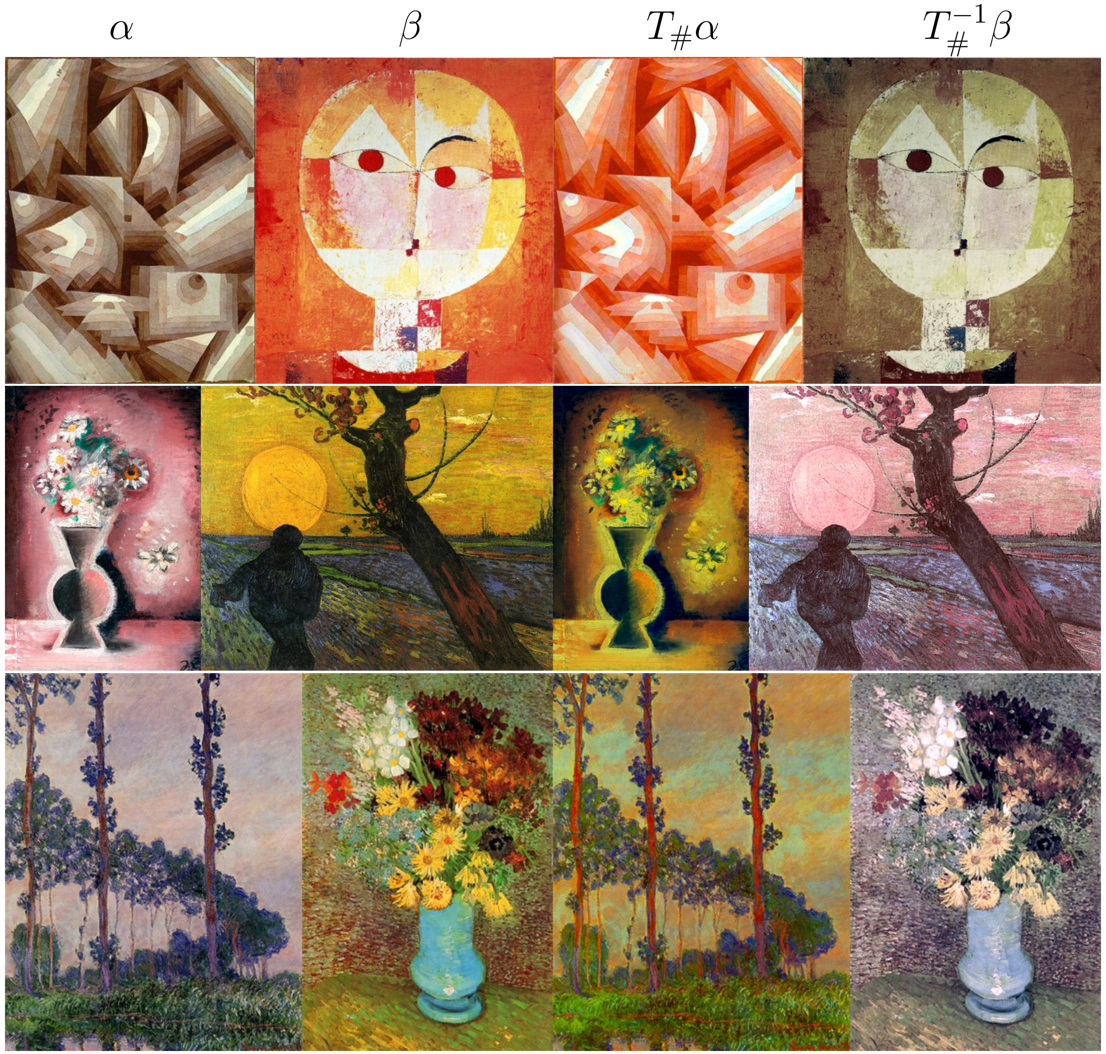




	Iter	Runtime (s)	Dual Value
Meta OT + W2GN	None	$3.5 \cdot 10^{-3} \pm 2.7 \cdot 10^{-4}$	$0.90 \pm 6.08 \cdot 10^{-2}$
	1k	$0.93 \pm 2.27 \cdot 10^{-2}$	$1.0 \pm 2.57 \cdot 10^{-3}$
	2k	$1.84 \pm 3.78 \cdot 10^{-2}$	$1.0 \pm 5.30 \cdot 10^{-3}$
W2GN	1k	$0.90 \pm 1.62 \cdot 10^{-2}$	$0.96 \pm 2.62 \cdot 10^{-2}$
	2k	$1.81 \pm 3.05 \cdot 10^{-2}$	$0.99 \pm 1.14 \cdot 10^{-2}$

More Meta OT color transfer predictions





This talk: amortized optimization for OT

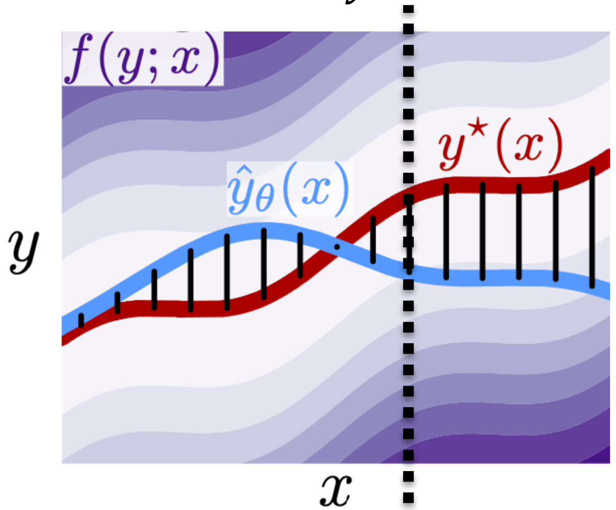
Amortizing the Kantorovich dual (Meta Optimal Transport)

$$\hat{\psi}(\alpha, \beta, c) \in \operatorname*{argsup}_{\psi \in L^1(\alpha)} \int_{\mathcal{Y}} \psi^c(y) d\beta(y) - \int_{\mathcal{X}} \psi(x) d\alpha(x)$$

Amortizing the c-transform (the convex conjugate)

$$\psi^c(y) \stackrel{\text{def}}{=} \inf_x \psi(x) + c(x, y)$$

$$y^\star(x) \in \operatorname*{argmin}_y f(y; x)$$



Vertical slices are optimization problems

Solving Euclidean Wasserstein-2 problems

Monge problem (primal)

$$T^\star(\alpha,\beta) \in \operatorname*{argmin}_{T\in\mathcal{C}(\alpha,\beta)} \mathbb{E}_{x\sim\alpha}\|x-T(x)\|_2^2$$

Kantorovich dual

$$\hat{f} \in \operatorname*{argmax}_{f\in\mathcal{L}^1(\alpha)} -\mathbb{E}_{x\sim\alpha}[f(x)] - \mathbb{E}_{y\sim\beta}[f^\star(y)]$$

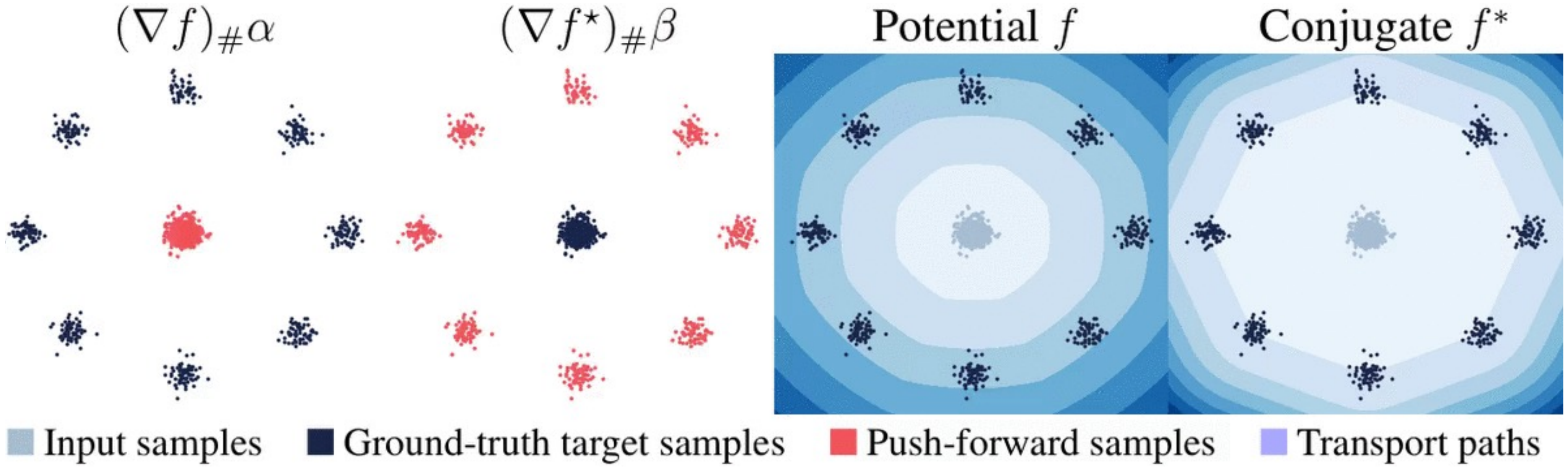
***c*-transform** becomes the **convex conjugate**

$$f^\star(y) := -\inf_{x\in\mathcal{X}} J_f(x;y) \quad \text{with objective} \quad J_f(x;y) := f(x) - \langle x, y\rangle.$$

Brenier's theorem gives $T^\star = \nabla\hat{f}$

Solve by **parameterizing** f_θ with an MLP and **optimizing the dual**

Computing the conjugate is hard, so **amortize the conjugate**



Learning the dual potentials

2-wasserstein approximation via restricted convex potentials with application to improved training for GANs. Taghvaei and Jalali, 2019.

Parameterize the potential potential $f_\theta \colon \mathcal{X} \to \mathbb{R}$

Optimize the dual objective

$$\max_\theta \mathcal{V}(\theta) \quad \text{where} \quad \mathcal{V}(\theta) := -\mathop{\mathbb{E}}_{x\sim\alpha}[f_\theta(x)] - \mathop{\mathbb{E}}_{y\sim\beta}[f_\theta^\star(y)] = -\mathop{\mathbb{E}}_{x\sim\alpha}[f_\theta(x)] + \mathop{\mathbb{E}}_{y\sim\beta}[J_{f_\theta}(\breve{x}(y))].$$

$$J_f(x;y) := f(x) - \langle x, y\rangle.$$

Assumes access to the **exact** conjugate is available

Differentiating and applying Danskin's envelope theorem gives:

$$\begin{aligned}\nabla_\theta \mathcal{V}(\theta) &= \nabla_\theta \left[-\mathop{\mathbb{E}}_{x\sim\alpha}[f_\theta(x)] + \mathop{\mathbb{E}}_{y\sim\beta}[J_{f_\theta}(\breve{x}(y))]\right] \\ &= -\mathop{\mathbb{E}}_{x\sim\alpha}[\nabla_\theta f_\theta(x)] + \mathop{\mathbb{E}}_{y\sim\beta}[\nabla_\theta f_\theta(\breve{x}(y))]\end{aligned}$$

Objective-based amortization of the conjugate

Three-Player Wasserstein GAN via Amortised Duality. Nhan Dam et al., IJCAI 2019.
Optimal transport mapping via input convex neural networks. Makkuva et al., ICML 2020.

Predict the solution to the conjugate with a model $\tilde{x}_\varphi$
Learn to optimize the conjugate objective everywhere it will be sampled (across β)

$$\min_{\varphi} \mathcal{L}_{\text{obj}}(\varphi) \text{ where } \mathcal{L}_{\text{obj}}(\varphi) := \mathop{\mathbb{E}}_{y \sim \beta} J_f(\tilde{x}_\varphi(y); y).$$
$$J_f(x; y) := f(x) - \langle x, y \rangle.$$

Replace the exact conjugate with the amortized prediction in the dual:

$$\max_{\theta} \min_{\varphi} \mathcal{V}_{\text{MM}}(\theta, \varphi) \text{ where } \mathcal{V}_{\text{MM}}(\theta, \varphi) := -\mathop{\mathbb{E}}_{x \sim \alpha} [f_\theta(x)] + \mathop{\mathbb{E}}_{y \sim \beta} [J_{f_\theta}(\tilde{x}_\varphi(y); y)].$$

Amortizing the conjugate with cycle consistency

Wasserstein-2 generative networks. Korotin et al., ICLR 2020.

Predict the solution to the conjugate with a model $\tilde{x}_\varphi$
Learn to optimize the conjugate objective everywhere it will be sampled (across β)

Taking the **optimality conditions of the conjugate** result in a **cycle consistency term**

$$J_f(x; y) := f(x) - \langle x, y \rangle. \qquad \nabla_x J_f(x; y) = \nabla_x f(x) - y = 0$$

$$\min_\varphi \mathcal{L}_{\rm cycle}(\varphi) \text{ where } \mathcal{L}_{\rm cycle}(\varphi) := \mathop{\mathbb{E}}_{y \sim \beta} \|\nabla_x J_f(\tilde{x}_\varphi(y); y)\|_2^2 = \mathop{\mathbb{E}}_{y \sim \beta} \|\nabla_x f(\tilde{x}_\varphi(y)) - y\|_2^2.$$

Replace the exact conjugate with the amortized prediction in the dual

Fine-tuning and regression

On amortizing convex conjugates for optimal transport. Amos, 2022.

Extremely easy to **fine-tune a prediction** with Adam or L-BFGS
Gives a much more stable estimation for the dual objective

Algorithm 2 CONJUGATE(f, y, x_{init})

$x \leftarrow x_{\text{init}}$
while unconverged **do**
  Update x with $\nabla_x J_f(x; y)$
end while
return optimal $\breve{x}(y) = x$

Amortize by regressing onto the fine-tuned prediction:

$$\min_{\varphi} \mathcal{L}_{\text{reg}}(\varphi) \text{ where } \mathcal{L}_{\text{reg}}(\varphi) := \mathop{\mathbb{E}}_{y \sim \beta} \|\tilde{x}_\varphi(y) - \breve{x}(y)\|_2^2.$$

The right amortization choices are important

On amortizing convex conjugates for optimal transport. Amos, 2022.

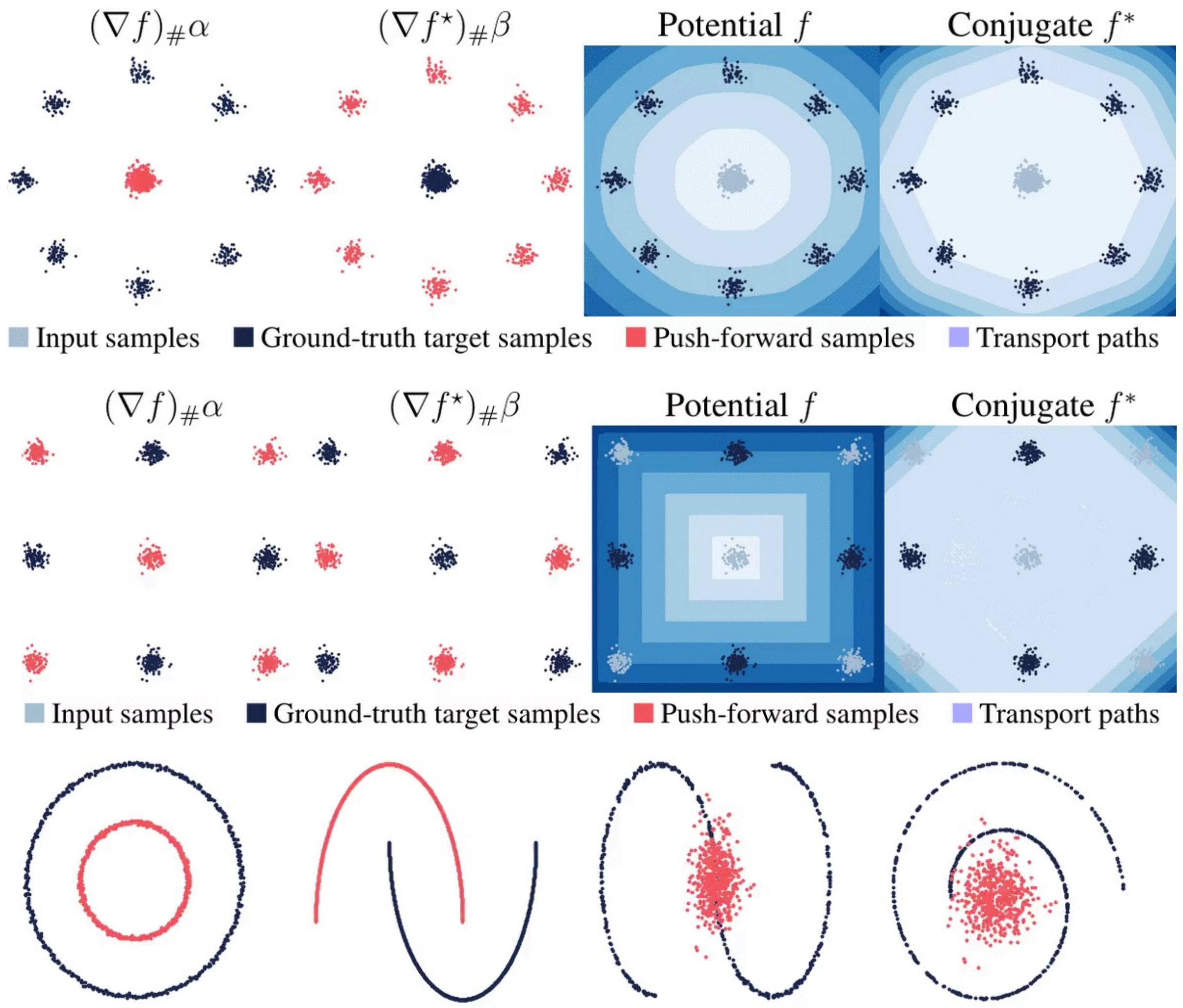
Results on the Wasserstein 2 benchmark (NeurIPS 2021)

Evaluation metric: unexplained variance percentage

Potential model: the non-convex neural network (MLP) described in app. B.4 **Amortization model:** the MLP described in app. B.2

Amortization loss	Conjugate solver	$D=2$	$D=4$	$D=8$	$D=16$	$D=32$	$D=64$	$D=128$	$D=256$
Cycle	None	0.05 ±0.00	0.35 ±0.01	1.51 ±0.08	>100	>100	>100	>100	>100
Objective	None	>100	>100	>100	>100	>100	>100	>100	>100
Cycle	L-BFGS	>100	>100	>100	>100	>100	>100	>100	>100
Objective	L-BFGS	0.03 ±0.00	0.22 ±0.01	0.60 ±0.03	0.80 ±0.11	2.09 ±0.31	2.08 ±0.40	0.67 ±0.05	0.59 ±0.04
Regression	L-BFGS	0.03 ±0.00	0.22 ±0.01	0.61 ±0.04	0.77 ±0.10	1.97 ±0.38	2.08 ±0.39	0.67 ±0.05	0.65 ±0.07
Cycle	Adam	0.18 ±0.03	0.69 ±0.56	1.62 ±2.82	>100	>100	>100	>100	>100
Objective	Adam	0.06 ±0.01	0.26 ±0.02	0.63 ±0.07	0.81 ±0.10	1.99 ±0.32	2.21 ±0.32	0.77 ±0.05	0.66 ±0.07
Regression	Adam	0.22 ±0.01	0.28 ±0.02	0.61 ±0.07	0.80 ±0.10	2.07 ±0.38	2.37 ±0.46	0.77 ±0.06	0.75 ±0.09
Improvement factor over prior work		**3.3**	**3.1**	**3.0**	**1.8**	**2.7**	**1.5**	**3.0**	**4.4**

	Amortization loss	Conjugate solver	Potential Model	Early Generator	Mid Generator	Late Generator
*[W2]	Cycle	None	ConvICNN64	1.7	0.5	0.25
*[MM]	Objective	None	ResNet	2.2	0.9	0.53
*[MM-R†]	Objective	None	ResNet	1.4	0.4	0.22
	Cycle	None	ConvNet	>100	26.50 ±60.14	0.29 ±0.59
	Objective	None	ConvNet	>100	0.29 ±0.15	0.69 ±0.90
	Cycle	Adam	ConvNet	0.65 ±0.02	0.21 ±0.00	0.11 ±0.04
	Cycle	L-BFGS	ConvNet	0.62 ±0.01	0.20 ±0.00	0.09 ±0.00
	Objective	Adam	ConvNet	0.65 ±0.02	0.21 ±0.00	0.11 ±0.05
	Objective	L-BFGS	ConvNet	0.61 ±0.01	0.20 ±0.00	0.09 ±0.00
	Regression	Adam	ConvNet	0.66 ±0.01	0.21 ±0.00	0.12 ±0.00
	Regression	L-BFGS	ConvNet	0.62 ±0.01	0.20 ±0.00	0.09 ±0.01
			Improvement factor over prior work	**2.3**	**2.0**	**2.4**



Learning flows via the Kantorovich dual

Challenges for learning flows (with potentials or otherwise)
1. The model needs to be invertible
2. The likelihood of the base density is required

$$p_Y(y) = p_X(f^{-1}(y)) \left| \frac{\partial f^{-1}(y)}{\partial y} \right|$$

Optimizing the potential-based flow for the **Kantorovich dual** can help with both of these!
1. Often parameterize the model as a non-convex MLP, invertibility no longer matters
2. Only requires samples from the densities

$$\max_\theta \mathcal{V}(\theta) \quad \text{where} \quad \mathcal{V}(\theta) := -\mathop{\mathbb{E}}_{x\sim\alpha}[f_\theta(x)] - \mathop{\mathbb{E}}_{y\sim\beta}[f_\theta^\star(y)] = -\mathop{\mathbb{E}}_{x\sim\alpha}[f_\theta(x)] + \mathop{\mathbb{E}}_{y\sim\beta}[J_{f_\theta}(\breve{x}(y))].$$

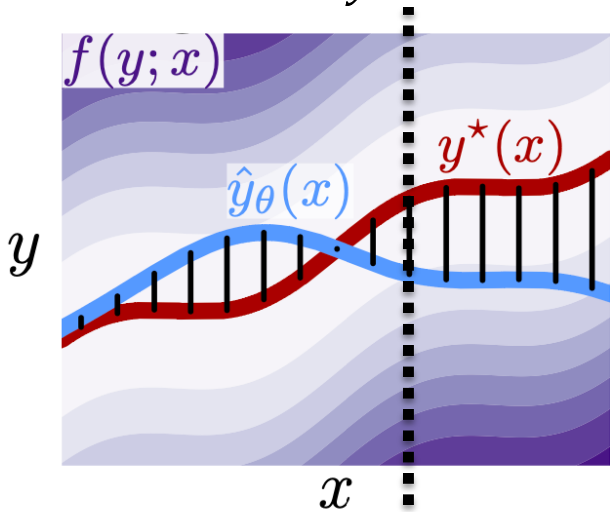
$$J_f(x; y) := f(x) - \langle x, y \rangle.$$

Conclusions

Amortized optimization foundations are here!
Useful for the **optimal transport dual** or ***c*-transform**

The **amortized prediction** does **not** need to be highly accurate
Can easily **check optimality conditions** and **fine-tune**

$$y^\star(x) \in \operatorname*{argmin}_y f(y;x)$$



Vertical slices are optimization problems

Amortized optimization for computing optimal transport maps

Brandon Amos • Meta AI (FAIR) NYC

http://github.com/bamos/presentations

Tutorial on amortized optimization, Brandon Amos, Foundations and Trends in ML, to appear.
Meta Optimal Transport, Brandon Amos, Samuel Cohen, Giulia Luise, Ievgen Redko, 2022.
On amortizing convex conjugates for optimal transport, Brandon Amos, 2022.